

PATROL[®] Knowledge Module for Event Management User Guide

Version 2.5

January 20, 2003



Copyright 2003 BMC Software, Inc., as an unpublished work. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc. All other registered trademarks or trademarks belong to their respective companies.

THE USE AND CONTENTS OF THIS DOCUMENTATION ARE GOVERNED BY THE SOFTWARE LICENSE AGREEMENT ENCLOSED AT THE BACK OF THIS DOCUMENTATION.

Restricted Rights Legend

U.S. GOVERNMENT RESTRICTED RIGHTS. UNPUBLISHED—RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b), or DFARS 227.7202, as amended from time to time. Contractor/Manufacturer is BMC Software, Inc., 2101 CityWest Blvd., Houston, TX 77042-2827, USA. Any contract notices should be sent to this address.

Contacting BMC Software

You can access the BMC Software Web site at <http://www.bmc.com>. From this Web site, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

Address BMC Software, Inc.
2101 CityWest Blvd.
Houston TX 77042-2827

Telephone 713 918 8800 or
800 841 2031

Fax 713 918 8000

Outside United States and Canada

Telephone (01) 713 918 8800

Fax (01) 713 918 8000

Customer Support

You can obtain technical support by using the Support page on the BMC Software Web site or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, please see “Before Contacting BMC Software.”

Support Web Site

You can obtain technical support from BMC Software 24 hours a day, 7 days a week at **<http://www.bmc.com/support.html>**. From this Web site, you can

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by Telephone or E-mail

In the United States and Canada, if you need technical support and do not have access to the Web, call 800 537 1813. Outside the United States and Canada, please contact your local support center for assistance. To find telephone and e-mail contact information for the BMC Software support center that services your location, refer to the Contact Customer Support section of the Support page on the BMC Software Web site at **www.bmc.com/support.html**.

Before Contacting BMC Software

Before you contact BMC Software, have the following information available so that Customer Support can begin working on your problem immediately:

- product information
 - product name
 - product version (release number)
 - license number and password (trial or permanent)
- operating system and environment information
 - machine type
 - operating system type, version, and service pack or other maintenance level such as PUT or PTF
 - system hardware configuration
 - serial numbers
 - related software (database, application, and communication) including type, version, and service pack or maintenance level

- sequence of events leading to the problem
- commands and options that you used
- messages received (and the time and date that you received them)
 - product error messages
 - messages from the operating system, such as `file system full`
 - messages from related software

Contents

About This Book	xiii
Chapter 1	Product Components and Capabilities
Features	1-2
Architecture	1-2
Application Classes and Instances	1-4
Application Class Hierarchy	1-5
Application InfoBox Items	1-5
Chapter 2	Getting Started
Requirements	2-3
System	2-3
License	2-3
Logon Account and Default PATROL Account	2-4
PATROL Security	2-4
Installation	2-6
Planning	2-6
Installation Options	2-9
Typical Installation	2-9
Custom Installation	2-12
Migration	2-15
Upgrading	2-15
Deleting Menu Commands from Windows Consoles	2-16
Deleting Menu Commands from Unix Consoles	2-17
How to Load and Unload Knowledge Modules	2-18
Loading Knowledge Modules	2-19
Unloading Knowledge Modules	2-23
Configuration Planning	2-26

Notification Servers	2-26
Notification Targets	2-27
Availability Targets and Monitors	2-28
Additional Alert Settings and Configuration Options	2-29
Configuration Tasks	2-29
Identifying and Editing Notification Scripts	2-31
Testing the Notification Script	2-34
Configuring the Notification Servers	2-36
Configuring Remote Agents	2-38
Adding Availability Targets	2-42
Configuring Availability Failover	2-44
Getting Started Tasks	2-45
Rewording Messages	2-46
Setting Parameter Thresholds	2-48
Setting Blackout Periods	2-53
Setting Notification Targets	2-59

Chapter 3 PATROL Objects, Configuration Variables, and Event Management Rules

PATROL Objects	3-2
Example: PATROL Object	3-3
Example: Rule Inheritance	3-4
Example: PATROL KM for Event Management E-mail Rule ..	3-5

Chapter 4 Menu Commands

COMPUTER Menu Commands	4-2
Quick Config	4-2
Alert Settings	4-3
Availability	4-18
Parameter Settings	4-22
Instance Filtering	4-24
Configuration DB	4-25
Reports	4-25
Admin	4-27
AS_EVENTSPRING Application Menu Commands	4-28
About	4-28
Manage Events	4-28
Reports	4-29
Alert Testing	4-29
Refresh Parameters	4-29

	AS_AVAILABILITY Application Menu Commands	4-30
	About	4-30
	Alert Testing	4-30
	Refresh Parameters	4-31
Chapter 5	Parameters	
	Parameter Summary	5-2
	Parameter Defaults	5-3
Appendix A	Command-line Interface	
Appendix B	PATROL KM for Event Management Reference	
	Alert and Notification Settings	B-2
	Remote Notification Settings	B-8
	Notification Server Settings	B-9
	Availability Monitor Settings	B-10
	Parameter Settings	B-12
	Application Class Settings	B-14
	Menu Command Access Settings	B-15
Appendix C	Sample Scenarios	
	Scenario: Send E-mail Notification for Alarm or Warning State . . .	C-1
	Assumptions	C-1
	Where to Start	C-2
Glossary		
Index		

Figures

Figure 1-1	Sample PATROL KM for Event Management Environment . . .	1-3
Figure 2-1	Select Products and Components to Install Dialog Box for Typical Windows Installation	2-11
Figure 2-2	Select Products and Components to Install Dialog Box for Custom Installation (Windows Example)	2-14
Figure 2-3	Quick Config - Notification Server Dialog Box	2-37
Figure 2-4	Notification Server Settings Dialog Box	2-39
Figure 2-5	Primary Notification Server Settings Dialog Box	2-40
Figure 2-6	Backup Notification Server Settings Dialog Box	2-41
Figure 2-7	Availability Monitor Add Target Dialog Box	2-42
Figure 2-8	Choose Primary Monitor Dialog Box	2-45
Figure 2-9	Set Event Management Alert Variables/Rules Dialog Box	2-46
Figure 2-10	. Set Event Management KM Variables/Rules (Choose the Target Classes) Dialog Box	2-48
Figure 2-11	Set Event Management KM Variables/Rules (Choose the target Instances) Dialog Box	2-49
Figure 2-12	Set Event Management KM Variables/Rules (Choose target Parameters) Dialog Box	2-49
Figure 2-13	Configure Thresholds Dialog Box	2-50
Figure 2-14	Set Event Management Blackout Variables/Rules (Choose the Target Classes) Dialog Box	2-54
Figure 2-15	Set Event Management Blackout Variables/Rules (Set Blackout Times) Dialog Box	2-55
Figure 2-16	Availability - Target Dialog Box	2-57
Figure 2-17	AvailabilityMonitor (Set Blackout Times) Dialog Box	2-58
Figure 2-18	Set Event Management KM Variables/Rules (Choose the Target Classes) Dialog Box	2-60

Figure 2-19	Set Event Management KM Variables/Rules (Choose target Instances) Dialog Box	2-61
Figure 2-20	Set Event Management KM Variables/Rules (Choose target Parameters) Dialog Box	2-61
Figure 2-21	Set Event Management Alert Variables/Rules (Set Event Management Alert Variables) Dialog Box	2-62
Figure 4-1	Computer Menu Commands.	4-2
Figure 4-2	PARAMETER SETTINGS REPORT Dialog Box.	4-26
Figure 4-3	AS_EventSpring Application Menu Commands	4-28
Figure 4-4	AS_AVAILABILITY Application Menu Commands	4-30
Figure C-1	Scenario Send E-mail Notification for Alarm or Warning State Configuration	C-2
Figure C-2	QUICK CONFIG - NOTIFICATION SERVER Dialog Box	C-3
Figure C-3	NOTIFICATION SERVER SETTINGS Dialog Box	C-5
Figure C-4	Primary Notification Server Settings Dialog Box	C-5
Figure C-5	Message Rewording Dialog Box	C-8

Tables

Table 1-1	Application Classes and KM Files.	1-4
Table 1-2	AS_EVENTSPRING InfoBox Items.	1-6
Table 1-3	AS_AVAILABILITY InfoBox Items.	1-6
Table 2-1	Characteristics of Typical and Custom Installation	2-12
Table 2-2	Predefined Configuration Components (same for all consoles)	2-20
Table 2-3	Notification Server Configuration Information	2-26
Table 2-4	Notification Target Configuration Information	2-27
Table 2-5	Availability Targets	2-28
Table 2-6	Alert Settings and Configuration Options	2-29
Table 2-7	Accessing KM Menu Commands	2-30
Table 2-8	Notification Script Arguments.	2-35
Table 2-9	Notification Server Properties	2-37
Table 2-10	Notification Server Properties	2-40
Table 2-11	Availability Target Properties.	2-43
Table 2-12	Threshold Settings	2-51
Table 2-13	Blackout Properties	2-56
Table 3-1	PATROL Objects Hierarchy.	3-2
Table 3-2	Example: PATROL Object.	3-3
Table 3-3	Example: Rule Inheritance.	3-4
Table 3-4	PATROL KM for Event Management Variable Definition	3-5
Table 4-1	Alert Actions	4-3
Table 4-2	Alert Actions Configuration	4-3
Table 4-3	Notification System	4-4
Table 4-4	Notification System Configuration	4-4
Table 4-5	Local Alert Settings: Alert Resend Configuration.	4-5
Table 4-6	Local Alert Settings: Notification Command Configuration. . .	4-6
Table 4-7	Local Alert Settings: Recovery Action Command Configuration	4-6

Table 4-8	Local Alert Settings: Recovery Action Command Type Configuration 4-7
Table 4-9	Local Alert Settings: Send Reset On Init Configuration 4-7
Table 4-10	Remote Alert Settings: Configure Notification Servers Configuration 4-8
Table 4-11	Remote Alert Settings: Remote Comm Settings Configuration. 4-8
Table 4-12	Notification Targets: Email Target Configuration 4-10
Table 4-13	Notification Targets: Pager Target Configuration. 4-10
Table 4-14	Notification Targets: Custom Target Configuration 4-11
Table 4-15	Notification Targets: TT Target Configuration. 4-11
Table 4-16	Alert Messages Configuration 4-12
Table 4-17	Alert Messages Replacement Variables 4-12
Table 4-18	Blackout Periods Configuration 4-16
Table 4-19	Remote Target Setting Options. 4-17
Table 4-20	Notification Server Settings: Remote Target Setting Configuration 4-17
Table 4-21	Custom Identifiers Configuration. 4-17
Table 4-22	Overrides Configuration. 4-18
Table 4-23	Add Target Configuration. 4-18
Table 4-24	Add Target: Updated Flag Configuration. 4-19
Table 4-25	Failover Settings: Identify Primary Configuration. 4-20
Table 4-26	Availability: Blackout Periods Configuration 4-21
Table 4-27	Availability: Ping Command Configuration. 4-21
Table 4-28	Availability: Checker Account Configuration 4-22
Table 4-29	Parameter Settings: Thresholds Configuration. 4-22
Table 4-30	Parameter Settings: Polltimes Configuration 4-23
Table 4-31	Parameter Settings: Status Flags Configuration. 4-23
Table 4-32	Instance Filtering: Edit Filter List Configuration. 4-24
Table 4-33	Instance Filtering: Edit Filter List Configuration. 4-24
Table 4-34	Parameter Settings Report Formats 4-26
Table 5-1	PATROL KM for Event Management Parameters 5-2
Table 5-2	PATROL KM for Event Management Parameter Defaults. 5-3
Table B-1	Alert and Notification Settings. B-2
Table B-2	Remote Notification Settings B-8
Table B-3	Notification Server Settings B-9
Table 2-4	Availability Monitor Settings B-10
Table 2-5	Parameter Settings B-12
Table B-6	Application Class Settings B-14
Table B-7	Menu Command Access Settings. B-15

About This Book

This book contains detailed information about PATROL Knowledge Module for Event Management and is intended for system administrators and database administrators (DBAs). The KM is also referred to as PATROL KM for Event Management.

Note

This book assumes that you are familiar with your host operating system. You should know how to perform basic actions in a window environment, such as choosing menu commands and dragging and dropping icons.

How This Book Is Organized

This book is organized as follows. In addition, this book contains a glossary of terms and an index.

Chapter/Appendix	Description
Chapter 1, “Product Components and Capabilities”	provides an overview of the PATROL KM for Event Management
Chapter 2, “Getting Started”	provides information on installing, configuring, and getting started using the PATROL KM for Event Management
Chapter 3, “PATROL Objects, Configuration Variables, and Event Management Rules”	provides information on PATROL objects, configuration variables, and event management rules
Chapter 4, “Menu Commands”	describes the PATROL KM for Event Management menu commands
Chapter 5, “Parameters”	provides a listing of the PATROL KM for Event Management parameters and lists their default settings
Appendix A, “Command-line Interface”	provides information on the command-line interface that can be used to trigger a PATROL KM for Event Management NOTIFY_EVENT from scripts or batch programs
Appendix B, “PATROL KM for Event Management Reference”	provides a comprehensive listing of PATROL KM for Event Management settings and rules
Appendix C, “Sample Scenarios”	provides a sample scenario on setting up PATROL KM for Event Management to send an e-mail notification when PATROL goes into a warning or an alarm state

Related Documentation

BMC Software products offer several types of documentation:

- online and printed books
- online Help
- release notes

For information about installation, migration, configuration, and tuning, see the following documents:

Topic	Publication
migrating from an earlier version	<i>PATROL Migration Tools User Guide</i>
installing and upgrading	<i>PATROL Installation Reference Manual</i>
configuring and tuning	<i>PATROL Agent Reference Manual</i> <i>PATROL Console for Microsoft Windows User Guide, Volume 1</i> <i>PATROL Console for Microsoft Windows User Guide, Volume 2</i> <i>PATROL Console for Microsoft Windows User Guide, Volume 3</i> <i>PATROL Console for Unix User Guide</i> <i>PATROL Central Operator — Microsoft Windows Edition Getting Started</i> <i>PATROL Central Operator — Web Edition Getting Started</i>

To view the complete PATROL documentation library, visit the support page on the BMC Software Web site at **<http://www.bmc.com/support>**. Log on and select a product to access the related documentation. If you are a first-time user and have purchased a product, you can request a permanent user name and password by registering at the Customer Support page. If you are a first-time user and have *not* purchased a product, you can request a temporary user name and password from your BMC Software sales representative.

The complete PATROL documentation library is also available on the PATROL documentation CD that is included with major releases of the PATROL Console and Agent.

Online and Printed Books

The books that accompany BMC Software products are available in online and printed formats. Online books are formatted as Portable Document Format (PDF) files. Some online books are also formatted as HTML files.

To Access Online Books

To view any online book that BMC Software offers, visit the Customer Support page of the BMC Software Web site at **<http://www.bmc.com/support>**. You can also access PDF books from the documentation CD that accompanies your product.

Use the free Acrobat Reader from Adobe Systems to view, print, or copy PDF files. In some cases, installing the Acrobat Reader and downloading the online books is an optional part of the product-installation process. For information about downloading the free reader, go to the Adobe Systems Web site at **<http://www.adobe.com>**.

To Request Additional Printed Books

BMC Software provides some printed books with your product order. To request additional books, go to **<http://www.bmc.com/support>**.

Online Help

You can access Help for a product through the product's **Help** menu. The Help provides information about the product's graphical user interface (GUI) and provides instructions for completing tasks.

Release Notes

Printed release notes accompany each BMC Software product. Release notes provide up-to-date information such as

- updates to the installation instructions
- last-minute product information

The latest versions of the release notes are also available on the Web at <http://www.bmc.com/support>.

Conventions

The following conventions are used in this book:

- This book includes special elements called *notes*, *warnings*, *examples*, and *tips*:

Note

Notes provide additional information about the current subject.

Warning

Warnings alert you to situations that can cause problems, such as loss of data, if you do not follow instructions carefully.

Example

An example clarifies a concept discussed in text.

Tip

Tips contain information that might improve product performance or that might make procedures easier to follow.

- All syntax, operating system terms, and literal examples are presented in this typeface.
- In instructions, **boldface** type highlights information that you enter. File names, directories, Web addresses, and e-mail addresses also appear in boldface type.

- The symbol => connects items in a menu sequence. For example, **Actions => Create Test** instructs you to choose the **Create Test** command from the **Actions** menu.
- The symbol >> denotes one-step instructions.
- In syntax, path names, or system messages, *italic* text represents a variable, as shown in the following examples:

The table *tableName* is not available.

system/instance/fileName

- In syntax, the following additional conventions apply:
 - A vertical bar (|) separating items indicates that you must choose one item. In the following example, you would choose *a*, *b*, or *c*:

a | b | c
 - An ellipsis (. . .) indicates that you can repeat the preceding item or items as many times as necessary.
 - Square brackets ([]) around an item indicate that the item is optional.

Product Components and Capabilities

This chapter provides an overview of the PATROL Knowledge Module for Event Management. The Knowledge Module is also referred to as PATROL KM for Event Management.

Features	1-2
Architecture	1-2
Application Classes and Instances	1-4
Application Class Hierarchy	1-5
Application InfoBox Items	1-5

Features

PATROL KM for Event Management provides event notification (by using e-mail or integrating with paging systems), event filtering, message rewording, and centralized alert management features. The KM addresses issues such as scalability, reliability, ease of use, integration points, and implementation strategies. Features include

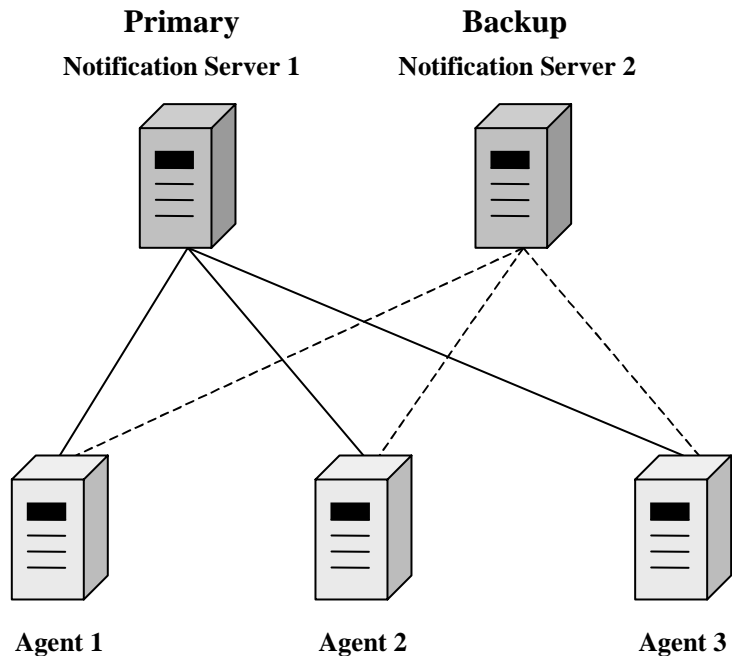
- highly scalable architecture with redundancy and automatic fail-over
- flexible alert settings
 - notification procedures
 - recovery actions
 - alert message rewording and translation
 - blackout periods
 - notification targets (for example, e-mail or paging systems)
 - parameter thresholds and polling times
- ability to change KM settings without changing BMC Software or third-party PATROL KMs
- checking the availability of agents and hosts
- integration with any paging solution, compiled executable, or script. Sample scripts are provided with the KM
- simplified and lower cost of integrating enterprise event consoles and tools

Architecture

The PATROL KM for Event Management architecture consists of monitored hosts (remote agents) and notification servers. Notification servers collect and process events from monitored systems according to defined notification rules. Notification rules include who gets notified, message rewording, and blackouts.

Both the monitored systems and the notification servers require a PATROL Agent and the PATROL KM for Event Management. Figure 1-1 on page 1-3 shows an PATROL KM for Event Management environment with several monitored hosts and two notifications servers (a primary and a backup).

Figure 1-1 Sample PATROL KM for Event Management Environment



The PATROL KM for Event Management architecture does not require a notification server to perform notifications such as paging and e-mails. Any monitored host running the PATROL KM for Event Management can be configured to perform its own notifications, event filtering, and alert control.

In the typical configuration, monitored hosts forward events to a notification server. When primary and backup notification servers exist, the backup notification server is used only when the primary notification server is unavailable. Failover is automatic. In the event that both primary and backup notification servers are unavailable, events are queued at the local agent. When communications are restored, all queued events are forwarded to the notification servers.



Application Classes and Instances

The PATROL KM for Event Management contains two application classes:

- AS_EVENTSPRING
- AS_AVAILABILITY

Table 1-1 describes the PATROL KM for Event Management application classes, icons and KM files.

Table 1-1 Application Classes and KM Files

Application Class	KM File	Description
 EVENT MANAGEMENT	AS_EVENTSPRING.km	menu commands, parameters, and InfoBox items for managing and reporting on events managed by the KM
 AS_AVAILABILITY	AS_AVAILABILITY.km	instances, menu commands, parameters, and InfoBox items for monitored hosts and agents

Application Class Hierarchy

The AS_EVENTSPRING application class icon resides at the application level beneath the computer icon. The AS_EVENTSPRING application class has only a single instance which is Event Management.

The AS_AVAILABILITY icon represents the Availability application class and resides at the application level beneath the computer icon. All monitored hosts appear beneath the Availability container and are accessed by double-clicking the AS_AVAILABILITY icon.

Application InfoBox Items

The status of an application instance can be reviewed by accessing the InfoBox for the application instance. To access an InfoBox, use the following procedure that applies to your environment:

PATROL Console for Unix

1. Double-click the application icon under the computer icon to view the application instance.
2. Using the middle mouse button, click the instance icon.

PATROL Console for Windows NT

1. Double-click on the application icon under the computer icon to view the application instance.
2. Right-click the instance icon to display a pop-up menu.
3. Choose **InfoBox**.

Table 1-2 shows a list of the InfoBox items for the AS_EVENTSPRING application class.

Table 1-2 AS_EVENTSPRING InfoBox Items

Item	Description
KM Version	PATROL KM for Event Management Version number
Spool Directory	directory used to store output files generated by parameter reports and recovery actions executed by the PATROL KM for Event Management

Table 1-3 shows a list of the InfoBox items for the application instances of the AS_AVAILABILITY application class.

Table 1-3 AS_AVAILABILITY InfoBox Items

Item	Description
Primary Monitor	agent with primarily responsibility for performing availability checks
Ping Command	command used to perform ping checks
Ping Host (ICMP)?	indicates whether the indicated host is being pinged using the ICMP protocol
Ping PATROLAgent?	indicates whether the PATROL Agent is being checked on the indicated host
Ping SNMP Agent?	indicates whether the SNMP agent is being checked on the indicated host
Blacked Out?	indicates whether the selected instance is currently being blacked out

Getting Started

This chapter provides information on installing and configuring PATROL KM for Event Management.

Requirements	2-3
System	2-3
License	2-3
Logon Account and Default PATROL Account	2-4
PATROL Security	2-4
Installation	2-6
Planning	2-6
Installation Options	2-9
Typical Installation	2-9
Custom Installation	2-12
Migration	2-15
Upgrading	2-15
How to Load and Unload Knowledge Modules	2-18
Loading Knowledge Modules	2-19
Unloading Knowledge Modules	2-23
Configuration Planning	2-26
Notification Servers	2-26
Notification Targets	2-27
Availability Targets and Monitors	2-28
Additional Alert Settings and Configuration Options	2-29
Configuration Tasks	2-29
Identifying and Editing Notification Scripts	2-31
Testing the Notification Script	2-34
Configuring the Notification Servers	2-36
Configuring Remote Agents	2-38

Adding Availability Targets	2-42
Configuring Availability Failover.	2-44
Getting Started Tasks	2-45
Rewording Messages	2-46
Setting Parameter Thresholds.	2-48
Setting Blackout Periods	2-53
Setting Notification Targets	2-59

Requirements

Before installing the PATROL KM for Event Management, verify that your environment meets all of the requirements needed to install and operate the PATROL KM for Event Management. The following sections describe the system, license, logon, and security requirements.

System

To use the PATROL KM for Event Management your computer must meet the following requirements:

- hardware
 - RAM - 250 KB
 - hard disk - 500 KB
 - software
 - PATROL 3.3 or later
 - operating system
 - all Unix/Linux platforms supported by PATROL
 - Windows NT 4.0 Server (with SP5 or higher)
 - Windows 2000 Server (with SP2 or higher)
 - OpenVMS V7.0 or greater (limited support)
- OpenVMS systems can not be used as notification servers or availability monitors, but they can send notifications to Unix or Windows notification servers, and they can be monitored by Unix or Windows availability monitors.

License

Verify that you have a valid demonstration license (typically good for 30 days) or permanent license to run your PATROL products. If you have not yet installed a permanent license, contact your BMC Software sales representative or the BMC Software Contract Administration department for licensing information.

Logon Account and Default PATROL Account

During installation of the PATROL Agent, you are asked to specify the default PATROL account. This user account is automatically assigned the rights needed to operate PATROL. BMC Software recommends that you create a unique user account for the PATROL default. If one does not already exist on your computer, see the PATROL installation guide for your operating system for instructions.

Warning

Do not use the administrator account to install PATROL products. Create a separate account that has the system administrator privileges. If you use the administrator account, files created by PATROL will be owned by the administrator, resulting in possible security or file access problems.

PATROL Security

You can secure the data passed between PATROL components and restrict unauthorized users from accessing your data by implementing PATROL security. PATROL now contains five security policy levels with predefined security configurations that you can select when you install PATROL. You can install the least secure or the most secure features of PATROL, depending on your system needs and the complexity of securing your systems.

Note

Review the security level definitions in the *PATROL Security User Guide* before installing PATROL to determine the appropriate security level for your system needs.

If you want to implement a new security level after having previously installed security, you must uninstall your current implementation of PATROL and reinstall it with the new security level.

High security requires more configuration of the communicating components (the agent, console, and console server) and is more difficult to use than lower levels of security. You can select the security level that best balances the ease of use with your need for security.

The lowest level (0) is a minimal level of security with no configuration requirements. At the highest security level (4), all communicating components must authenticate with each other and key databases must validate connection requests.

Note

All components in a system, including agents, consoles and console servers, must operate at the same level of security to communicate with each other. This requirement is ensured when you install PATROL with the lowest level of security (the default level of 0).

For more information about implementing and using PATROL security, see the *PATROL Security User Guide*.

How PATROL Security Affects KMs

PATROL Security is installed as part of the agent, console, and console server. KMs inherit the security policy from the agent and console on which they are installed.

Note

For the PATROL Knowledge Module for Event Management, the security level of the remote agents and the notification server must be the same.

Installation

Install the PATROL KM for Event Management on all PATROL Agent computers that you want to monitor for PATROL events. You must also install the KM on any PATROL Classic Consoles, console servers, or PATROL Central Operator - Web Edition Web server that is used to configure the KM or view PATROL KM for Event Management information.

Note

PATROL KM for Event Management modifies the PATROL Standard Event Catalog. Installing the PATROL KM for Event Management archives then overwrites the **StdEvents.ctg** file. If you have made changes to **StdEvents.ctg**, create a backup copy of the file before installing PATROL KM for Event Management.

The **StdEvents.ctg** file is located in the **PATROL_HOME/lib/knowledge** directory. See the *PATROL Installation Reference Manual* for more information on the PATROL Standard Events Catalog.

BMC Software recommends that you install the product on a limited number of development or test computers first, configure and test the BMC product, and then install it onto production computers.

You can install the PATROL KM for Event Management product on multiple computers, and the product configuration can be varied depending your needs. The following sections outline what to plan for and consider before you install PATROL KM for Event Management.

Planning

The installation plan is simple and straightforward: install the PATROL KM for Event Management on managed systems where you will manage PATROL events.

The installation program provides the following options:

- install to the local computer now
- create an installable image on the local computer that you can then install on the local computer *or on other computers* at any time

For additional information about installing from an installable image, refer to the *PATROL Installation Reference Manual*.

Note

You can export the installation package after you have made all of the installation selections. If you place that exported installation package in a shared directory, you can use that same installation package to install BMC Software products on all computers that share the same BMC Software products directory, PATROL default login, PATROL Agent port number, PATROL 3.x and 7.x product directories, and security options.

The installation program shipped with PATROL 3.5 and PATROL 7 prompts you to select the roles performed by the computer onto which you are installing BMC Software products. Before beginning the installation process, review the following definitions of the roles presented in the installation program, and decide which of these describes the roles that each computer in your system performs. During installation, select one or all of the following options:

- **Console Systems** (formerly referred to as console computers and client application system) host user desktop applications such as consoles, user interfaces, viewers, and browsers. Select this option if you are installing on a computer that will perform any of the following roles:
 - monitor and manage on the Web using a PATROL Central Operator - Web Edition console (PATROL 7) (Unix or Windows)
 - monitor and manage on Windows using a PATROL Central Operator - Windows Edition console (PATROL 7)
 - monitor, manage, and develop KMs on Unix or Windows using a PATROL Console for Unix or PATROL Console for Windows (PATROL 3)

- **Managed Systems** (formerly referred to as agent computers) host software that manages the resources on the system, such as a PATROL Agent, PATROL Knowledge Modules, and Service Reporting Retrievers. Select this option if you are installing on a computer that will perform any of the following roles:
 - host a PATROL Agent 3.4 or greater (works with both the PATROL 3.x and PATROL 7.x architecture)
 - host a PATROL Central Operator - Microsoft Windows Edition (PATROL 7) console
 - host KMs and components that contain the knowledge PATROL uses to monitor the resources on this computer
- **Common Services** (new with PATROL 7.1) host services that are shared among managed systems and client application systems. Each of these common services can be installed on any computer on the network. Select this option if you are installing on a computer that will perform any of the following roles:
 - host the PATROL Central Operator - Web Edition (PATROL 7) console
 - host the RTserver
 - host the PATROL Console Server

Installation Options

Use one of the following options to install PATROL KM for Event Management:

- The **Typical** installation option installs a predefined set of components (selectable entities). To install components that are not among the predefined set, see “Custom Installation” on page 2-12.

Note

BMC Software recommends that you use the Typical installation if you do not yet have a good understanding of PATROL, and you are installing to a computer that does not have a prior version of PATROL.

Note

The Typical and Custom installations automatically preload the KMs listed in the **EVENT_MANAGEMENT.kml** file.

- The **Custom** installation option installs the components and KMs that you select. Use the custom installation if you have a good understanding of PATROL, or you are installing to a computer that has a prior version of PATROL.

Typical Installation

Use the following general procedure to perform a typical installation that includes the PATROL KM for Event Management:

- Step 1** From the installation CD, run **setup.exe** (Windows) or **setup.sh** (Unix).
- Step 2** In the Select Installation Option dialog box, select **I want to install products on this computer** or **I want to create an installable image to be installed later**.

2.A If you selected **I want to create an installable image to be installed later**, populate the destination path where you want to save the installable image by using the Browse button or text box.

Step 3 In the Select Type of Installation dialog box, select **Typical**.

Step 4 In the Specify Installation Directory dialog box, identify your BMC Software products installation directory.

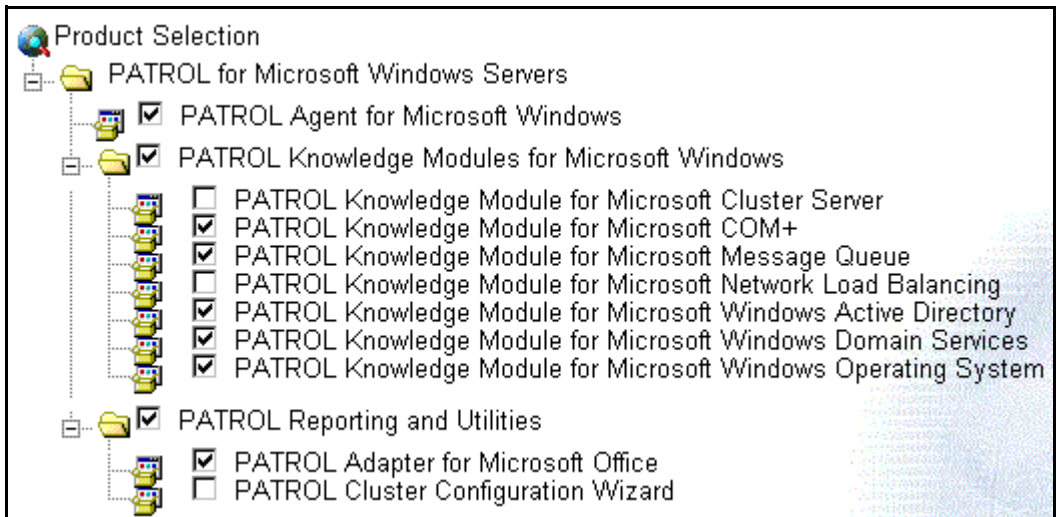
Step 5 In the Select System Roles dialog box, select one or more of the following:

- Select **Managed System** if you are installing to a computer that will host a PATROL Agent.
- Select **Console Systems** if you are installing to a computer that will host a PATROL Classic Console or PATROL Central Operator - Microsoft Windows Edition console.
- Select **Common Services** if you are installing to a computer that will host a PATROL Console Server, RTserver, or PATROL Central Operator - Web Edition.

Step 6 In the Select Products and Components to Install dialog box, select the products and components that you want to install.

See Figure 2-1 on page 2-11 for an example of this dialog box on a Windows installation.

Figure 2-1 Select Products and Components to Install Dialog Box for Typical Windows Installation



Note

The PATROL KM for Event Management does not appear in the Product Selection list, but it is installed by default with the typical installation option.

Step 7 In the Provide Information for the PATROL Agent dialog box, indicate whether you want to start the PATROL Agent automatically after completing the installation.

This dialog box opens only if you selected **Managed System** in Step 5 of this procedure.

Step 8 Complete the remaining dialog boxes.

Custom Installation

A custom installation is similar to the typical installation. The differences are shown in Table 2-1.

Table 2-1 Characteristics of Typical and Custom Installation

Characteristic	Typical	Custom
complexity	lower	higher
automatically preloads KMs	yes	yes
allows port number designation	no	yes
allows PATROL 3.x directory designation	no	yes
allows PATROL 7 directory designation	no	yes

Use the following general procedure to perform a custom installation:

- Step 1** From the installation CD, run **setup.exe** (Windows) or **setup.sh** (Unix).
- Step 2** In the Select Installation Option dialog box, select **I want to install products on this computer** or **I want to create an installable image to be installed later**.
- 2.A** If you selected **I want to create an installable image to be installed later**, populate the destination path where you want to save the installable image by using the Browse button or text box.
- Step 3** In the Select Type of Installation dialog box, select **Custom**.
- Step 4** In the Specify Installation Directory dialog box, identify your BMC Software products installation directory.
- Step 5** In the Select System Roles dialog box, select one or more of the following:
- Select **Managed System** if you are installing to a computer that will host a PATROL Agent.

- Select **Console Systems** if you are installing to a computer that will host a PATROL Console or PATROL Central Operator - Microsoft Windows Edition console.
- Select **Common Services** if you are installing to a computer that will host a PATROL Console Server, RTserver, or PATROL Central Operator - Web Edition.

Step 6 In the Select Products and Components to Install dialog box, select the products and components that you want to install.

See Figure 2-2 on page 2-14 for a Windows example.

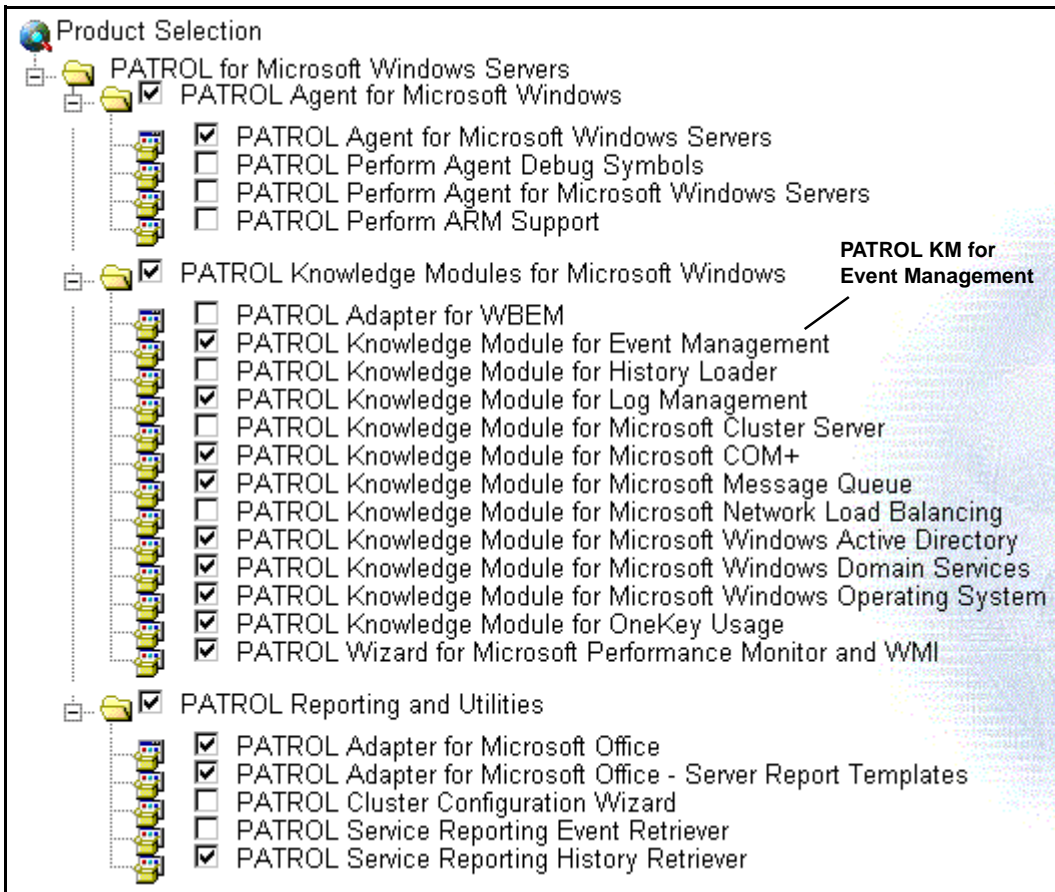
6.A In the Provide Information for the PATROL Agent dialog box, indicate whether you want to start the PATROL Agent automatically after completing the installation.

This dialog box opens only if you selected **Managed System** in Step 5 of this procedure.

6.B In the Provide the PATROL 3.x Product Directory dialog box, identify the BMC products installation directory and the PATROL 3.x product directory.

This dialog box opens only if you selected **Console Systems** in Step 5 of this procedure.

Figure 2-2 Select Products and Components to Install Dialog Box for Custom Installation (Windows Example)



6.C In the Provide the PATROL 7.x Product Directory, identify the BMC Software products installation directory and the PATROL 7.x product directory.

This dialog box opens only if you selected **Common Services** in Step 5 of this procedure.

Step 7 Complete the remaining dialog boxes.

Note

The number and contents of the dialog boxes depend on your KM selections and your inputs to the dialog boxes. Click **Help**, as needed, to complete the dialog boxes.

Migration

The PATROL KM for Event Management 2.5.00 has no migration path from versions prior to 2.5.00. Migration of the PATROL KM for Event Management is not required because the configuration information, including parameter settings, is stored in the PATROL Agent configuration database and is backward compatible.

Upgrading

Although configuration information does not require migration, you must perform the following three tasks when you upgrade from a previous version of this KM to version 2.5.00:

- Remove Customized KM Files From the PATROL_CACHE
- Backup the Standard Event Catalog File
- Remove Existing Menu Commands

Remove Customized KM Files From the PATROL_CACHE

Before upgrading from a previous version of this KM, you must backup and remove all Event Management and EventSpring customized files from the PATROL_CACHE.

Backup the Standard Event Catalog File

Before upgrading, backup the catalog file **StdEvent.ctg** by renaming the file or copying it to another directory.

Remove Existing Menu Commands

Before upgrading from a previous version of this KM, you must remove existing menu commands that are installed in the ALL_COMPUTERS.KM.

You must perform the following steps from a PATROL Classic Developer Console. Use the procedure that applies to your operating system.

Deleting Menu Commands from Windows Consoles

Summary: You must perform the following steps from a PATROL Classic Developer Console before upgrading the KM.

Step 1 From the PATROL Classic Developer Console, select the **KM** tab.

Step 2 Expand the following folders in the KM tree view:

Knowledge Module => Computer Classes => ALL_COMPUTERS => Menu Commands.

Step 3 Right-click **EVENTSPRING** or **Event Management**.

Step 4 Select **Delete**.

A dialog box asks you to confirm that you want to delete the command.

Step 5 Click **Yes**.

Step 6 Save your changes before you exit the PATROL Console.

Deleting Menu Commands from Unix Consoles

Summary: You must perform the following steps from a PATROL Classic Developer Console before upgrading the KM.

Step 1 From the PATROL Classic Developer Console main window, choose **Attributes => Computer Classes** from the menu bar.

The List of Computer Classes dialog box opens.

Step 2 Select the **ALL_COMPUTERS** class.

Step 3 Choose **Attributes => Commands => Menu Commands** from the **List of Computer Classes** menu.

The List of Menu Commands for the ALL_COMPUTERS class is displayed.

Step 4 Select **EVENTSPRING** or **Event Management**.

Step 5 Choose **Edit => Delete**.

A confirmation dialog box is displayed.

Step 6 Click **Yes**.

Step 7 Save your changes before you exit the PATROL Console.

How to Load and Unload Knowledge Modules

Installing PATROL KM for Event Management places the application files into the PATROL directory. You can load the files into the PATROL Console so that the PATROL KM for Event Management applications, commands, and parameters appear in the PATROL Console.

If you no longer want to use an application class that you previously loaded, you can use the unload instructions to unload the **.km** file so that its application class no longer appears in your console.

Loading Knowledge Modules

Summary: Before you can begin using KMs that you have installed, you must first load them with a PATROL Console. In this section, follow the instructions that apply to your console.

Loading KMs with PATROL Central - Windows Edition

PATROL Central - Windows Edition contains a Loading KMs Wizard that you can use to specify which KMs to load on which computers.

Step 1 On the **Common Tasks** taskpad, double-click the **Load Knowledge Modules** icon.

PATROL Central - Windows Edition displays the wizard.

Step 2 Click **Next** to start the wizard.

The wizard lists each computer that has a PATROL Agent installed.

Step 3 Select the check boxes for the computers on which you want to load KMs, and click **Next**.

The wizard displays a list of available **.kml** files for each computer selected in the previous step. Each **.kml** file is listed once for each computer. You can display **.km** files by changing the filter.

The KMs available in this product are listed in Table 2-2 on page 2-20.

Note

Unless you are an advanced PATROL user, use the **.kml** files to load product component files. Loading individual **.km** files can break the interdependencies between the **.km** files, while loading **.kml** files preserves these dependencies.

Table 2-2 Predefined Configuration Components (same for all consoles)

Product (.kml file)	Components (.km files)
EVENT_MANAGEMENT.kml	AS_AVAILABILITY.km AS_EVENTSPRING.km AS_EVENTSPRING_ALL_COMPUTERS.km

Step 4 Select the check boxes for the **.km** and computer pairs to load.

Step 5 Click **Next**, and click **Finish**.

PATROL loads the selected KMs on the selected computers.

Loading KMs with PATROL Central - Web Edition

PATROL Central - Web Edition has a Loading KMs feature that you can use to control which KMs to load on which computers.

Step 1 From the Monitored Systems page, click **Load/Unload KMs**.

The Load KMs page opens, listing each computer that has a PATROL Agent installed.

Step 2 Select the computers on which you want to load KMs, and click **Next**.

The Load KMs page displays a list of available **.km** and **.kml** files.

If you selected more than one computer, the only **.km** and **.kml** files that are listed are the ones that have been installed on all of the selected computers. If a particular **.km** or **.kml** file was installed only on one computer, you must choose that computer by itself to load the file.

The **.km** files available in this product are listed in Table 2-2 on page 2-20.

Note

Unless you are an advanced PATROL user, use the **.kml** files to load product component files. Loading individual **.km** files can break the interdependencies between the **.km** files, while loading **.kml** files preserves these dependencies.

Step 3 Select the **.km** or **.kml** files that you want to load.

Step 4 Click **Finish**.

PATROL loads the selected KMs on the selected computers.

Note

If you want to load a **.km** or **.kml** file that was not listed in Step 2, ensure that the KM is installed on the appropriate computer, and select only that computer in Step 2.

Loading KMs with the PATROL Classic Console for Windows

Step 1 From the PATROL Classic Console for Windows menu, choose **File => Load KM**.

The Load KMs dialog box displays a list of available **.kml** files. The KMs available in this product are listed in Table 2-2 on page 2-20.

Note

Unless you are an advanced PATROL user, use the **.kml** files to load product component files. Loading individual **.km** files can break the interdependencies between the **.km** files, while loading **.kml** files preserves these dependencies.

Step 2 Select one or more of the **.kml** files, and click **Open**.

PATROL loads the selected KMs on all of the computers listed under PATROLMainMap.

Loading KMs with the PATROL Classic Console for Unix

Step 1 From the PATROL Console for Unix menu, choose **File => Load KM**.

The Load KMs dialog box displays a list of available **.kml** files. The KMs available in this product are listed in Table 2-2 on page 2-20.

Note

Unless you are an advanced PATROL user, use the **.kml** files to load product component files. Loading individual **.km** files can break the interdependencies between the **.km** files, while loading **.kml** files preserves these dependencies.

Step 2 Select one or more of the **.kml** files, and click **Open**.

PATROL loads the selected KMs on all of the computers connected to your console.

Unloading Knowledge Modules

Summary: If you no longer want to use an application class that you previously loaded, you can unload the **.km** file so that its application class will no longer appears in your console.

The PATROL Classic Consoles refer to unloading as *deleting*. When you unload or delete a **.km** file using a console, the file is not deleted from the **patrol\knowledge** directories on the PATROL Console or the PATROL Agent computers.

Use the procedure that applies to your console.

Unloading KMs with PATROL Central - Windows Edition

PATROL Central - Windows Edition has a wizard that you can use to unload specified **.km** files from specified computers.

Step 1 On the Common Tasks taskpad, double-click the **Unload Knowledge Modules** icon.

PATROL Central - Windows Edition displays the wizard.

Step 2 Click **Next** to start the wizard.

The wizard lists each computer that has a PATROL Agent installed.

Step 3 Select the check boxes for the computers from which you want to unload **.km** files, and click **Next**.

The wizard displays a list of application class names (that correspond to **.km** file names) for each computer selected. Each application class name is listed once for each computer.

Step 4 Select the check boxes for the **.km** and computer pair that you want to unload, and click **Next**.

Step 5 Click **Finish**.

The console removes the selected **.km** files from the current management profile.

Unloading KMs with PATROL Central - Web Edition

PATROL Central - Web Edition has a feature that you can use to unload specified **.km** files from specified computers.

Step 1 From the Managed Systems page, click **Load/Unload KMs**.

The Load KMs page opens, listing each computer that has a PATROL Agent installed.

Step 2 Select the computers from which you want to unload **.km** files, and click **Next**.

The Load KMs page displays a list of **.km** files. Currently loaded **.km** files are highlighted in the list.

Step 3 Cancel the selection of the **.km** files that you want to unload.

Step 4 Click **Finish**.

The console removes the specified **.km** files from the current management profile.

Unloading KMs with the PATROL Classic Console for Windows

Unloading a KM is also referred to as *deleting* a KM in the PATROL Classic Console for Windows.

Step 1 From the **KM** tab, right-click the application class name that you want to delete, and choose **Delete** from the pop-up menu.

The console displays a dialog box that asks if you want to delete the selected application class.

Step 2 Click **Yes** to delete the application class.

The application class is removed from your cache directory and your console session file.

Step 3 Repeat Step 1 and Step 2 for all of the application classes (.km files) associated with the KM (.kml file) that you want to remove.

Step 4 From the console main menu, choose **File => Save KM** to save your changes.

Unloading KMs with the PATROL Classic Console for Unix

Unloading a KM is also referred to as *deleting* a KM in the PATROL Classic Console for Unix.

Step 1 From the main menu, choose **Attributes => Application Classes**.

The console displays the List of Application Classes dialog box.

Step 2 Click the name of the application class that you want to delete.

The console highlights the application class name.

Step 3 From the List of Application Classes menu, choose **Edit => Delete**.

The application class is removed from your cache directory and your console session file. The console removes the application class name from the List of Application Classes.

Step 4 Repeat Step 2 and Step 3 for all of the application classes (.km files) associated with the KM (.kml file) that you want to delete.

Step 5 From the List of Application Classes menu, choose **File => Save KM** to save your changes.

Configuration Planning

Before you can use PATROL KM for Event Management, you must gather information and plan your configuration of the KM. The following sections provide information on identifying the following information:

- notification servers
- remote notification targets
- availability targets and availability monitors
- additional alert settings and configuration options

Notification Servers

A notification server is the managed system that performs notifications and event collection on behalf of other PATROL Agents. You can define a primary and a backup notification server. When identifying a notification server, make certain that there will be no connectivity issues to these systems (such as firewalls) from the agents that it serves.

Complete the notification server information in Table 2-3:

Table 2-3 Notification Server Configuration Information

Notification Server	Primary	Backup
Hostname or IP Address		
PATROL Agent Port		
Username		
Password		
Protocol (TCP, UDP)		

For additional security, create an operating system account on the notification server systems to be used specifically for remote notification. This configuration avoids having to use the PATROL login, which may be common throughout your environment. The notification server login can be configured so that it is unable to fully login to the notification server system by using the operating system. For example, on Unix, give the notification server login an invalid login shell, such as `/bin/false`.

Note

You must install the PATROL Agent and the PATROL KM for Event Management on the notification server system.

Notification Targets

For alert notifications, you create a default e-mail account that is used as the default target for all PATROL objects that do not have a defined target.

To further configure and customize the PATROL KM for Event Management, you can specify e-mail targets for each PATROL object. For example, you may want Unix-related alerts to go to your Unix administrator and Windows-related alerts to go to your Windows administrator.

Create a notification table similar to the one in Table 2-4 to identify who to notify for each type of alert. Make sure that all targets exist, for example e-mail accounts and pagers.

Table 2-4 Notification Target Configuration Information

PATROL Object	E-mail Target	Paging Target
/	patrol@any.co.com	none
/CPU	unixadmin@any.co.com	unixadmin
/NT_CPU	ntadmin@any.co.com	ntadmin

Availability Targets and Monitors

You can ensure that your PATROL Agents are running by using the availability feature of the PATROL KM for Event Management. You can use the availability feature to define managed systems to monitor and to specify the computers to use as the primary and backup availability monitors.

The primary monitors checks the availability of the managed systems. The backup monitor checks the availability of the primary monitor. If the primary monitor goes offline, the backup monitor assumes the monitoring of the managed systems. When the primary monitor becomes available again, it resumes monitoring the managed systems, and the backup monitor resumes monitoring the primary monitor.

Complete the availability target information in Table 2-5, and specify one managed system as the primary availability monitor and one managed system as the backup availability monitor.

Table 2-5 Availability Targets

Managed System Hostname or IP Address	Agent Port	Primary or Backup

Additional Alert Settings and Configuration Options

Table 2-6 describes additional configuration settings to consider when configuring PATROL KM for Event Management.

Table 2-6 Alert Settings and Configuration Options

Option	Description
Rewording of Alert Messages	configure alert messages For example, you can reword FSCapacity alerts as follows: Filesystem /tmp on db_server is 99% full.
Blackout Periods	identify times when notification is suspended for particular PATROL objects or availability targets For example, during periods of scheduled downtime such as routine maintenance.

Configuration Tasks

This section describes the tasks required to configure the PATROL KM for Event Management. The following tasks are described in this section:

- Identifying and Editing Notification Scripts
- Testing the Notification Script
- Configuring the Notification Servers
- Configuring Remote Agents
- Adding Availability Targets
- Configuring Availability Failover

Before you begin the configuration of the PATROL KM for Event Management, review the following information:

- Verifying that PATROL KM for Event Management is Loaded
- Accessing KM Application Class Menus

Verifying that PATROL KM for Event Management is Loaded

To ensure that a managed system has the PATROL Knowledge Module for Event Management application loaded, ensure that each managed system displays the Event Management application class icon when viewed with a PATROL Console that has the PATROL Knowledge Module for Event Management loaded.

Note

The AS_AVAILABILITY application class icon is not displayed in the PATROL Console until availability targets are added.

Accessing KM Application Class Menus

You access KM application class menus differently in each PATROL Console.

Table 2-7 Accessing KM Menu Commands

Console	Method
PATROL Classic Console for Microsoft Windows	Right-click the application or computer icon, and choose KM Commands .
PATROL Classic Console for Unix	Right-click the application or computer icon.
PATROL Central Operator - Windows Edition	In the navigation pane, right-click a managed system or application icon, and choose Knowledge Module Commands from the pop-up menu.
PATROL Central Operator - Web Edition	In the tree view area, right-click a managed system, application class, or application instance, and choose Knowledge Module Commands from the pop-up menu.

Identifying and Editing Notification Scripts

Summary: The PATROL KM for Event Management uses notification scripts that call command-line utilities to initiate notification (such as e-mail and paging). In this task, you identify and edit the notification script that you will use to send notifications with the PATROL KM for Event Management.

Identifying and Editing the Required Notification Script

The PATROL KM for Event Management provides notification scripts in the **psl** directory that you can use with no modification for e-mail notifications with Blat on Windows or mailx on Unix. The PATROL KM for Event Management also includes a Perl script that you can use for e-mail on Windows or Unix with minor modifications.

Note

BMC Software recommends using the **.bat** or **.sh** scripts. To use the **.bat** script on Windows, you must download Blat from the Web. Ensure that Blat is installed, configured, and tested before you test the notification script.

The following sections describe how to use and modify the notification scripts to send e-mail notification.

Note

On Linux, you must change references to **mailx** in the script to **mail**.

Windows

On Windows the PATROL KM for Event Management uses the following script with Blat to send e-mail notifications.

%PATROL_HOME%\lib\psl\AS_EVSLocalAlertNotify.bat

Blat is a free, third-party utility that can be downloaded from the Web.

Note

The **AS_EVSLocalAlertNotify.bat** script looks for Blat in the following directory: **C:\Blat**

If you have Blat installed in another location, you must move Blat to this location or edit the script to execute Blat from the directory where you have it installed.

Unix

On Unix the PATROL KM for Event Management uses the following script with mailx to send e-mail notifications:

\$PATROL_HOME/lib/psl/AS_EVSLocalAlertNotify.sh

All Platforms Supporting Perl 5

Note

BMC Software recommends using the **.bat** or **.sh** scripts; however, you can use a Perl script if you prefer.

The PATROL KM for Event Management includes a Perl script that you can use to send e-mail notifications on Unix or Windows systems with Perl 5 installed.

Unix

On Unix the PATROL KM for Event Management uses the following script with mailx to send e-mail notifications:

\$PATROL_HOME/lib/psl/AS_EVSLocalAlertNotify.pl

Windows

On Windows the PATROL KM for Event Management uses the following Perl script with Blat to send e-mail notifications.

%PATROL_HOME%\lib\psl\AS_EVSLocalAlertNotify.pl

On Windows, you must modify the Perl script before you can use it to send e-mail notifications with Blat.

» Find the following line in the Perl script and remove the comment (#):

```
#system("c:\\blat\\blat.exe $email_file -t \"$ntargets\" -s \"$nmsg\"");
```

Note

The **AS_EVSLocalAlertNotify.pl** script expects Blat to be installed in the following directory: **C:\Blat**

If you have Blat installed in another location, you must move Blat to this location or edit this line of script to execute Blat from the directory where you have it installed.

Other Types of Notification

You can also use the notification scripts that are included with the PATROL KM for Event Management to send other types of notifications. In addition to e-mail, the following notification types are supported by the PATROL KM for Event Management notification scripts:

- paging
- trouble tickets
- custom

The PATROL KM for Event Management notification scripts have clearly marked sections in the scripts where you can add scripting to support these notification types.

Testing the Notification Script

Summary: Before you continue configuring the PATROL KM for Event Management, you should test your notification script to verify that e-mail and any other notification services are properly configured. In this task, you test your notification script.

Testing Your Notification Script

To perform a test, run the appropriate notification script as shown in the following example:

```
AS_EVSLocalAlertNotify.ext type "targets" "message"
```

Note

Replace **.ext** with the file extension of your notification script (**.bat**, **.sh**, or **.pl**).

The notification script arguments are described in Table 2-8, “Notification Script Arguments,” on page 2-35.

Table 2-8 Notification Script Arguments

Argument	Description
<i>type</i>	<p>the notification type that you are testing</p> <p>The following notification types are supported:</p> <ul style="list-style-type: none">• email—sends an e-mail notification• page—sends a page notification• tt—sends a trouble ticket notification• custom—sends a custom notification <p>Note The page, tt, and custom notification types require you to customized the notification scripts.</p>
<i>targets</i>	<p>a list of targets</p> <p>The list of targets is a comma-separated list of values that is used as the target of the notification.</p> <p>Note Any spaces are converted to commas by the notification script.</p>
<i>message</i>	the notification or test message

The following example shows a test of the **AS_EVSLocalAlertNotify.bat** script on Windows:

```
AS_EVSLocalAlertNotify.bat email "patrol@bmc.com" "This is a test."
```

Note

Rename the notification script and place it in a directory outside of the PATROL installation directories. This step preserves any changes if you ever reinstall PATROL or PATROL KM for Event Management. Place the notification script in the same location on both the primary and backup notification servers.

Configuring the Notification Servers

Summary: In this task, you configure the notification servers for the PATROL KM for Event Management.

Before You Begin

Before configuring the PATROL KM for Event Management, see “Notification Servers” on page 2-26 for information on determining which managed systems you will define as your notification servers.

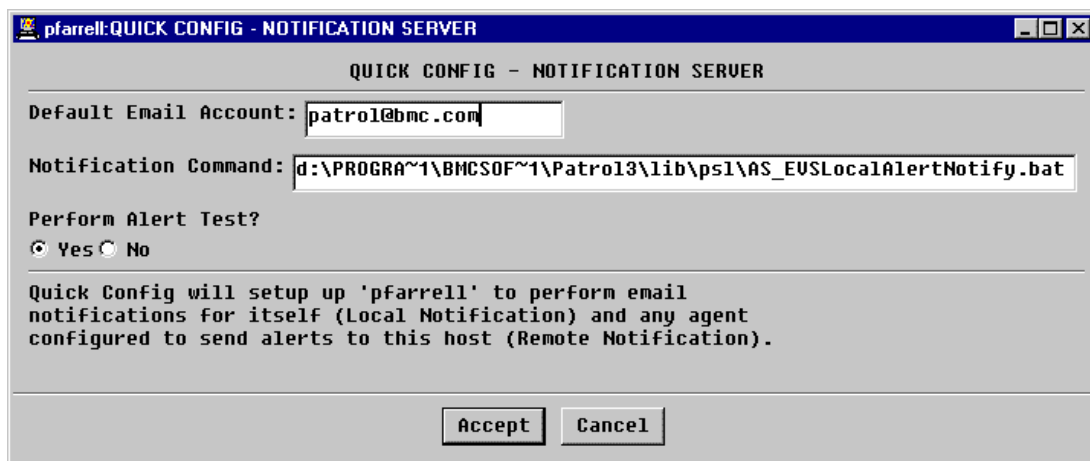
Configuring the Notification Servers

To configure the notification servers, perform the following steps using a PATROL Console.

- Step 1** Right-click the computer icon of the managed system you are using as your notification server.
- Step 2** Choose **KM Commands => Event Management => Quick Config => Notification Server**.

The Quick Config - Notification Server dialog box opens as shown in Figure 2-3 on page 2-37.

Figure 2-3 Quick Config - Notification Server Dialog Box



Use the Quick Config - Notification Server dialog box to specify the notification server properties. The notification server properties are described in Table 2-9:

Table 2-9 Notification Server Properties

Property	Description
Default Email Account	<p>the default e-mail address (notification target) that receives e-mails when an object goes into an alarm or warning state</p> <p>All events for PATROL objects that do not have defined notification targets are sent to this default e-mail address. See “Setting Notification Targets” on page 2-59 for information on setting notification targets.</p>
Notification Command	the complete path and filename of the notification script or command used to send notifications
Perform Alert Test	<p>whether you want to perform an alert test after the changes are accepted</p> <p>If this is your first time using the PATROL KM for Event Management, you should perform an alert test and verify that the notifications are received.</p>

Step 3 Define the notification server properties, and click **Accept**.

Step 4 Repeat this task (beginning on page 2-36) for the managed system you are using as the backup notification server.

Configuring Remote Agents

Summary: In this task you configure the remote PATROL Agents to use specified primary and backup notification servers.

Before You Begin

Configure and test the notification server setup before configuring the remote PATROL Agents.

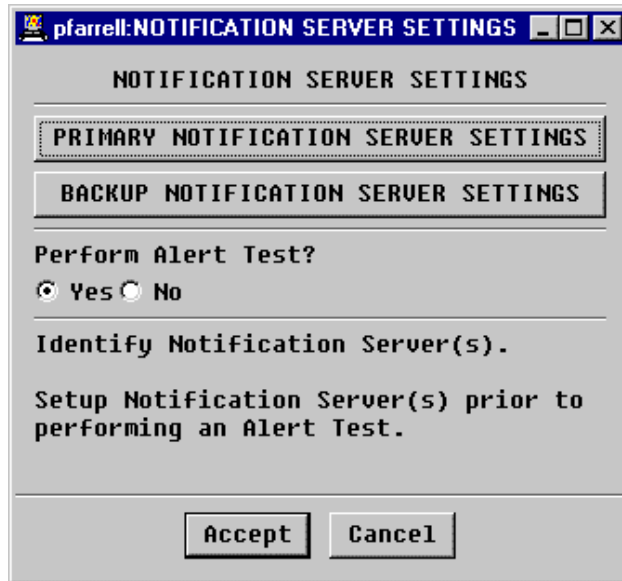
Configuring the Remote Agents

To configure the notification servers used by the remote agents, perform the following steps using a PATROL Console:

- Step 1** Right-click the computer icon of the managed system.
- Step 2** Choose **KM Commands => Event Management => Quick Config => Remote Agent**.

The Notification Server Settings dialog box opens as shown in Figure 2-4 on page 2-39.

Figure 2-4 Notification Server Settings Dialog Box

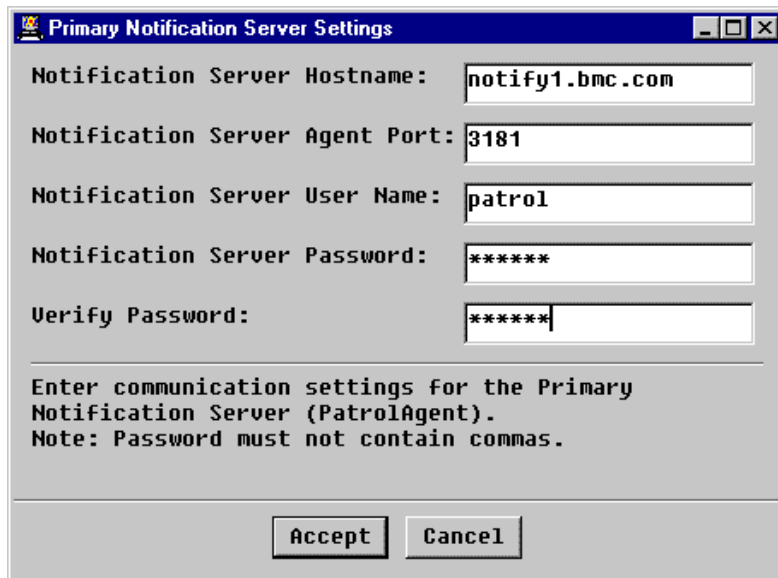


Use the Notification Server Settings dialog box to specify the primary and backup notification servers for the managed system.

Step 3 Click **Primary Notification Server Settings**.

The Primary Notification Server Settings dialog box opens as shown in Figure 2-5 on page 2-40.

Figure 2-5 Primary Notification Server Settings Dialog Box



The dialog box is titled "Primary Notification Server Settings". It contains five input fields with labels: "Notification Server Hostname:" (value: notify1.bmc.com), "Notification Server Agent Port:" (value: 3181), "Notification Server User Name:" (value: patrol), "Notification Server Password:" (value: *****, masked), and "Verify Password:" (value: *****, masked). Below the fields is a text area with the message: "Enter communication settings for the Primary Notification Server (PatrolAgent). Note: Password must not contain commas." At the bottom are two buttons: "Accept" and "Cancel".

Use the Primary Notification Server Settings dialog box to specify the properties of the primary notification server for the managed system. The notification server properties are described in Table 2-10:

Table 2-10 Notification Server Properties

Property	Description
Notification Server Hostname	the hostname of the primary notification server for the selected managed system
Notification Server Agent Port	the port number that the selected managed system will use to connect to the notification server
Notification Server User Name	the user name that the selected managed system will use to connect to the notification server
Notification Server Password	the password that the selected managed system will use to connect to the notification server
Verify Password	verify the password that the selected managed system will use to connect to the notification server

Step 4 Define the primary notification server properties, and click **Accept**.

Step 5 [Optional] Click **Primary Notification Server Settings**.

The Backup Notification Server Settings dialog box opens as shown in Figure 2-6:

Figure 2-6 Backup Notification Server Settings Dialog Box

Backup Notification Server Settings

Notification Server Hostname: notify2.bmc.com

Notification Server Agent Port: 3181

Notification Server User Name: patrol

Notification Server Password: *****

Verify Password: *****

Enter communication settings for the Backup Notification Server (PatrolAgent).
Note: Password must not contain commas.

Accept Cancel

Use the Backup Notification Server Settings dialog box to specify the properties of the backup notification server for the managed system. See Table 2-10, “Notification Server Properties,” on page 2-40 for more information on the notification server properties.

Step 6 Type the backup notification server properties, and click **Accept**.

Repeat this task (beginning on page 2-38) for each managed system to configure that system’s notification server setup.

Note

You can use the PATROL Configuration Manager to quickly configure all remote agents at once. See the *PATROL Configuration Manager User Guide* for more information.

Adding Availability Targets

Summary: An availability target defines a managed system that is monitored for availability. In this task you add an availability target.

Step 1 Right-click the computer icon of the managed system that will monitor the target.

Step 2 Choose **KM Commands => Event Management => Availability => Add Target**.
The Availability Monitor Add Target dialog box opens as shown in Figure 2-7:

Figure 2-7 Availability Monitor Add Target Dialog Box

AVAILABILITY MONITOR - Add Target

Hostname:

PATROL Agent Port:

SNMP Port:

DEFAULT SNMP SETTINGS

SNMP Community:

SNMP Timeout (ms):

SNMP Retries:

SNMP ObjectID:

Notes:

- * If a port is not entered, only the host will be checked with an ICMP ping.
- * The host/agent doing the monitoring must be able to resolve the entered hostname, otherwise, one or more ping failures will occur.
- * An ICMP ping will not be performed if an SNMP ping (get request) is selected for a particular target.

Use the Availability Monitor - Add Target dialog box to specify the properties of the availability target. Table 2-11, “Availability Target Properties,” on page 2-43 describes the availability target properties.

Table 2-11 Availability Target Properties

Property	Description
Hostname	<p>the host name or IP address of the managed system you are monitoring</p> <p>checks only the availability of this computer</p>
PATROL Agent Port	<p>the port number of the PATROL Agent on the managed system that you are monitoring</p> <p>checks the availability of the agent running on this computer</p> <p>Note: cannot use both PATROL Agent Port and SNMP Port</p>
SNMP Port	<p>the SNMP port number of the PATROL Agent on the managed system that you are monitoring</p> <p>If you are using SNMP to monitor a managed system, the HostPingFailures parameter is not used and it remains unavailable. The SnmpPingFailure parameter is used instead.</p> <p>checks the availability of the agent running on this computer</p> <p>Note: cannot use both PATROL Agent Port and SNMP Port</p>
SNMP Community	the SNMP community string of the managed system you are monitoring
SNMP Timeout	the SNMP connection timeout for the managed system you are monitoring
SNMP Retries	the number of times the PATROL Agent tries to connect to the managed system you are monitoring before it fails
SNMP Object ID	the SNMP ObjectID on the managed system you are monitoring

- Step 3** Define the availability target properties, and click **Accept**.
- Step 4** Repeat this task (beginning on page 2-42) for each managed system you want to monitor for availability.

Configuring Availability Failover

Summary: If you are using availability monitoring, configure availability monitoring failover. In this task, you configure availability monitoring failover.

Before You Begin

Before you configure availability monitoring failover, ensure that you have the managed systems that serve as the primary and backup availability monitors configured with the same availability targets.

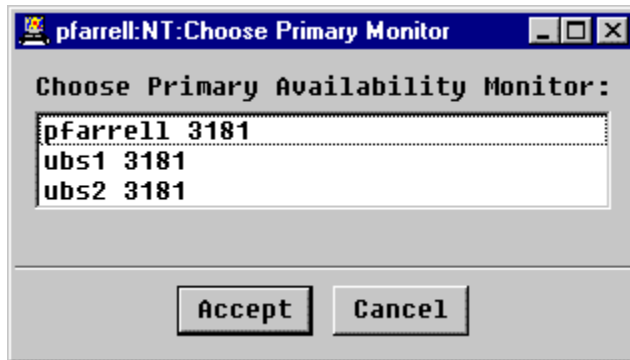
Note

You can define all the targets manually on one availability monitor (managed system) and then use the PATROL Configuration Manager to configure the other availability monitor. See the *PATROL Configuration Manager User Guide* for more information.

- Step 1** Right-click the computer icon of the managed system to use as the **backup** availability monitor.
- Step 2** Choose **KM Commands => Event Management => Availability => Failover Settings => Identify Primary**.

The Choose Primary Monitor dialog box opens as shown in Figure 2-8 on page 2-45.

Figure 2-8 Choose Primary Monitor Dialog Box



Step 3 Select the primary availability monitor, and click **Accept**.

The selected managed system is set as the primary availability monitor, and the managed system that you executed the command from in Step 1 becomes the backup availability monitor. The backup availability monitor only monitors the primary availability monitor. If the primary monitor becomes unavailable, the backup assumes monitoring until the primary monitor is available again.

Getting Started Tasks

This section describes the following basic tasks to get started using the PATROL KM for Event Management:

- Rewording Messages
- Setting Parameter Thresholds
- Setting Blackout Periods
- Setting Notification Targets

These tasks show just one example of many tasks that you can perform.

Rewording Messages

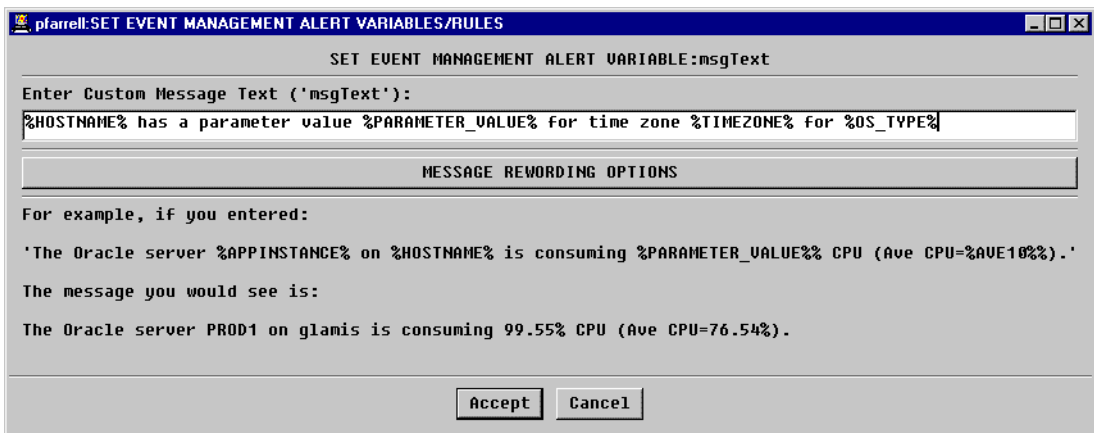
Summary: You can use the PATROL KM for Event Management to customize or reword alert messages. In this task, you reword the default message for all alerts.

Step 1 Right-click the computer icon of the managed system with the message that you want to reword.

Step 2 Choose **KM Commands => Event Management => Alert Settings => Alert Messages => Default Message Format**.

The Set Event Management Alert Variables/Rules dialog box opens as shown in Figure 2-9.

Figure 2-9 Set Event Management Alert Variables/Rules Dialog Box



Step 3 Type your reworded message in the Enter Custom Message Text text box.

You can use a combination of variables and text to create your reworded message. Click **Message Rewording Options** to see a description of the variables you can use to create your reworded message or see Table 4-17, “Alert Messages Replacement Variables,” on page 4-12 for a complete list and description of the message replacement variables provided by the PATROL KM for Event Management

Step 4 Click **Accept** to save your reworded message.

Setting Parameter Thresholds

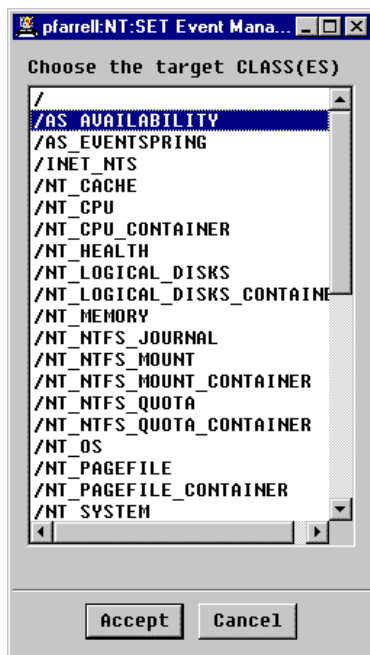
Summary: You can use the PATROL KM for Event Management to set PATROL parameter thresholds for any parameter on a managed system. In this task, you change the threshold for all instances of a PATROL parameter.

Step 1 Right-click the computer icon of a managed system.

Step 2 Choose **KM Commands => Event Management => Parameter Settings => Thresholds**.

The Set Event Management KM Variables/Rules (Choose the Target Classes) dialog box opens as shown in Figure 2-10.

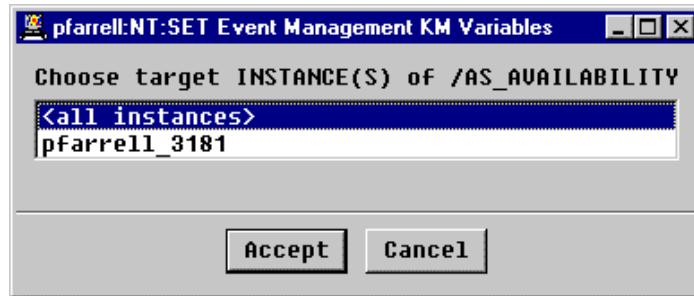
Figure 2-10 Set Event Management KM Variables/Rules (Choose the Target Classes) Dialog Box



Step 3 Select **/AS_AVAILABILITY** as the target class, and click **Accept**.

The Set Event Management KM Variables/Rules (Choose the target Instances) dialog box opens as shown in Figure 2-11.

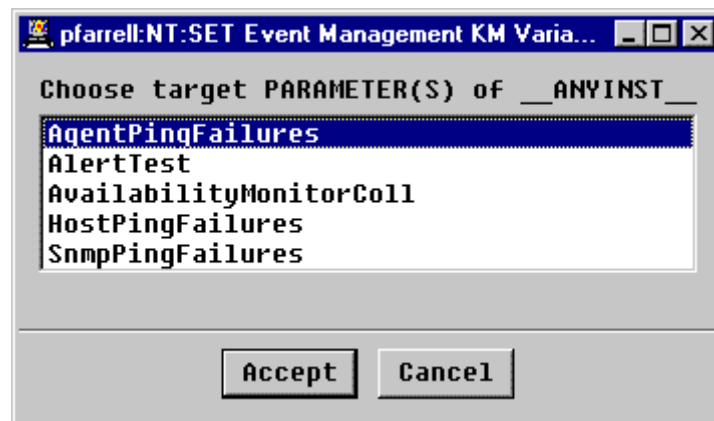
Figure 2-11 Set Event Management KM Variables/Rules (Choose the target Instances) Dialog Box



Step 4 Select **<all instances>**, and click **Accept**.

The Set Event Management KM Variables/Rules (Choose target Parameters) dialog box opens as shown in Figure 2-12.

Figure 2-12 Set Event Management KM Variables/Rules (Choose target Parameters) Dialog Box



Step 5 Select **AgentPingFailures** as the target parameter, and click **Accept**.

The Configure Thresholds dialog box opens as shown in Figure 2-13.

Figure 2-13 Configure Thresholds Dialog Box

Configure Thresholds
pfarrell:/AS_AVAILABILITY/_ANYINST_/AgentPingFailures

☒ **Active (Current settings from Active Parameter)**

Border Range:

☐ **Enable** **Alert State:** ☒ OK ☐ WARN ☐ ALARM

Range Settings: Min Max

Trigger Alarm: N:

Alarm Range 1:

☐ **Enable** **Alert State:** ☒ OK ☐ WARN ☐ ALARM

Range Settings: Min Max

Trigger Alarm: N:

Alarm Range 2:

☒ **Enable** **Alert State:** ☐ OK ☐ WARN ☒ ALARM

Range Settings: Min Max

Trigger Alarm: N:

Choose Duration of Change:

☒ **Apply to current session** ☒ **Make persistent**

Accept **Cancel**

Use the Configure Thresholds dialog box to specify the threshold settings of the selected objects. The threshold setting properties are described in Table 2-12:

Table 2-12 Threshold Settings

Setting	Description
Active	when selected, indicates that the parameter is active and the settings in the dialog box represent the current parameter settings
Border Range	<p>indicates the Border Range settings</p> <p>Click Enable to enable the border range; then specify border range properties</p> <p>Enable the border range if it is possible for the parameter to return a value outside of the other alarm range limits. You can use this setting primarily for information or as a third-level alert condition representing either a warning or an alarm state.</p>
Alarm Range 1	<p>indicates the Alarm Range 1 settings</p> <p>Select Enable to activate Alarm1; then specify a minimum and maximum range. Use Alarm1 as a first-level alert condition representing either a warning or an alarm state.</p> <p>Alarm1 values</p> <ul style="list-style-type: none"> • Must be less than Alarm2 values • Cannot overlap the Alarm2 range • Cannot fall outside the range limits or border range
Alarm Range 2	<p>indicates the Alarm Range 2 settings</p> <p>Select Enable to activate Alarm2; then specify a minimum and maximum range. Use Alarm2 as a second-level alert condition representing either a warning state or an alarm state.</p> <p>Alarm2 values</p> <ul style="list-style-type: none"> • Must be greater than Alarm1 values • Cannot overlap the Alarm1 range • Cannot fall outside the range limits or border range
Enable	makes the range active

Table 2-12 Threshold Settings

Setting	Description
Alert State	<p>Choose OK when the result of the range breach is informational for users or non-critical (for example, back up a file when it exceeds a certain size).</p> <p>Choose Warning or Alarm to have the parameter undergo a state change when the range is breached.</p>
Min	minimum value of acceptable range
Max	maximum value of acceptable range
Trigger Alarm	<p>determines when an alarm is triggered</p> <p>Select one of the following values:</p> <ul style="list-style-type: none"> • After an alarm has occurred “n” times Use this value to discount random spikes in the return values. You must also specify how many consecutive times the alarm range can be breached before the parameter actually alarms. • Immediately on alarm Use this value when you know the value returned is critical. • After all recovery actions fail Use this value if you want to be alerted after all recovery actions fail and the returned value is still within the defined warning or alarm range.
N	If you selected After an alarm has occurred “n” times for the trigger alarm, specify how many consecutive times you want the alarm value returned during parameter execution before the parameter actually alarms.
Apply to current session	<p>applies the selected parameter thresholds to the current agent session</p> <p>Note: Does not retain changes when the agent restarts.</p>
Make persistent	<p>retains the selected parameter thresholds</p> <p>Note: Retains changes when the agent restarts.</p>

Step 6 Define the threshold setting properties, and click **Accept**.

Setting Blackout Periods

Summary: Use the PATROL KM for Event Management to set blackout periods for PATROL objects and availability targets. A blackout period is a set time when alerts for the object are ignored. In this task, you set the blackout period of a PATROL object and an availability target.

About Blackout Periods

Blackout periods are set to prevent notification from taking place during a specified time period even if an alert condition occurs. You can set multiple blackout times per day.

Blackout periods apply to notification, and they can be applied to most PATROL objects. You can define blackout periods at the notification server or locally at the system where the alert occurs.

Local Blackout Periods

If a blackout is defined locally on a managed system, alerts are not forwarded to the remote notification server during the defined blackout period.

Notification Server Blackout Periods

If the blackout period is defined at the notification server, alerts are forwarded to the notification server during defined blackout periods, but the notification server does not perform notification, even though an alert is received.

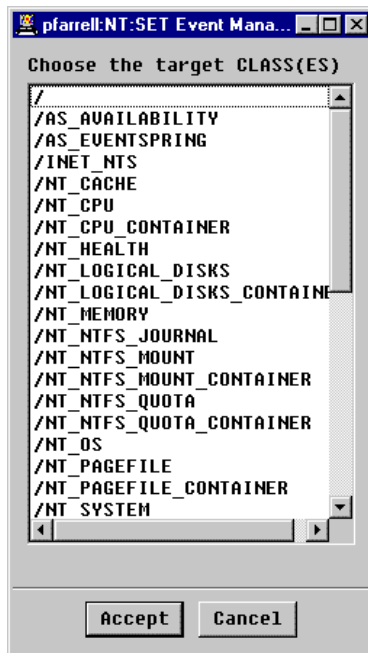
Setting an Alert Blackout for an Application Class

To set an alert blackout for an application class, perform the following steps:

- Step 1** Right-click the computer icon of a managed system.
- Step 2** Choose **KM Commands => Event Management => Alert Settings => Blackout Periods => Set for Classes**.

The Set Event Management Blackout Variables/Rules (Choose the Target Classes) dialog box opens as shown in Figure 2-14.

Figure 2-14 Set Event Management Blackout Variables/Rules (Choose the Target Classes) Dialog Box



- Step 3** Select an application class, and click **Accept**.

The Set Event Management Blackout Variables/Rules (Set Blackout Times) dialog box opens as shown in Figure 2-15 on page 2-55.

Figure 2-15 Set Event Management Blackout Variables/Rules (Set Blackout Times) Dialog Box

SET BLACKOUT TIMES

Current Blackout Times:
None Set

Blackout Start Time
00:00:00

Blackout Stop Time
00:00:00

Blackout Days
☐ Mon ☐ Tue ☐ Wed ☐ Thu ☐ Fri
☐ Sat ☐ Sun

☒ Merge Selected Blackout Periods with Pre-Existing Settings
☐ Replace Existing Blackout Periods with Current Selection

Blackout times apply to notification only.
When an alert condition occurs a NOTIFY_EVENT will be generated but notification will not be performed.
To prevent notifications and the NOTIFY_EVENT from being generated override the Alert Action, 'arsAction'.

Accept Cancel

Use this dialog box to specify the blackout properties for the selected application class. The blackout properties are described in Table 2-13, “Blackout Properties,” on page 2-56.

Table 2-13 Blackout Properties

Property	Description
Blackout Start Time	when the blackout begins
Blackout End Time	when the blackout stops
Blackout Days	the days of the week the blackout occurs
Merge Selected Blackout Periods with Pre-Existing Settings	merges the current blackout period with existing blackout periods for this object Note: You can use this feature to create multiple blackouts in a single day.
Replace Existing Blackout Periods with Current Selection	replaces all existing blackout periods with the defined blackout period

Example

To set a blackout from Friday at 19:00 until Sunday at 9:00 requires you to create three blackout periods and merge them as you create each one:

1. Blackout Start Time -->19:00:00
Blackout Stop Time --> 23:59:59
Blackout Days-->Fri
2. Blackout Start Time -->00:00:00
Blackout Stop Time --> 23:59:59
Blackout Days-->Sat
3. Blackout Start Time -->00:00:00
Blackout Stop Time --> 09:00:00
Blackout Days-->Sun

Step 4 Define the blackout properties, and click **Accept**.

Blackout periods for other objects are set using the same task, but you will choose the **Set for Instances** or **Set for Parameter** menu command in Step 2.

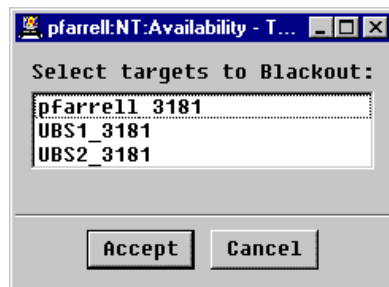
Setting an Availability Blackout

To set an availability blackout for a managed system, perform the following steps:

- Step 1** Right-click the computer icon of a managed system.
- Step 2** Choose **KM Commands => Event Management => Availability => Blackout Periods**.

The Availability - Target dialog box opens as shown in Figure 2-16.

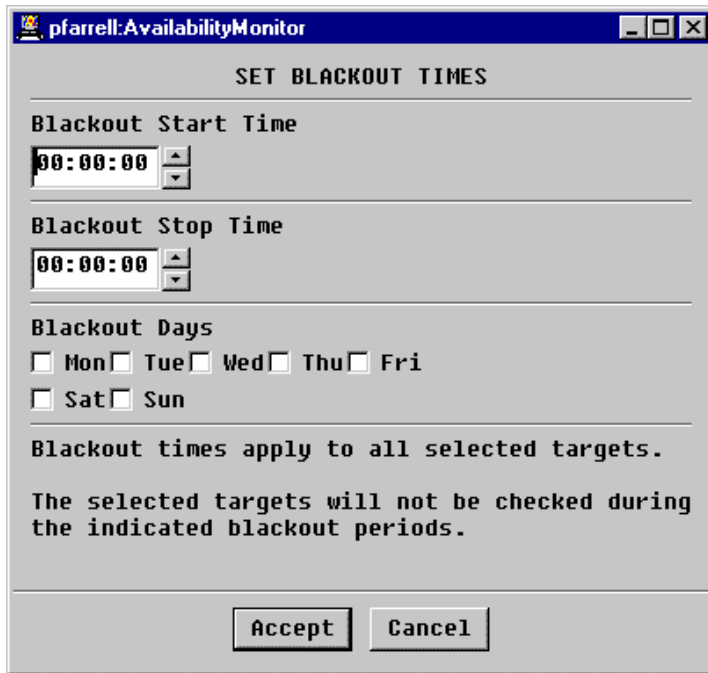
Figure 2-16 Availability - Target Dialog Box



- Step 3** Select one or more targets, and click **Accept**.

The AvailabilityMonitor (Set Blackout Times) dialog box opens as shown in Figure 2-17 on page 2-58.

Figure 2-17 AvailabilityMonitor (Set Blackout Times) Dialog Box



SET BLACKOUT TIMES

Blackout Start Time
00:00:00

Blackout Stop Time
00:00:00

Blackout Days
☐ Mon ☐ Tue ☐ Wed ☐ Thu ☐ Fri
☐ Sat ☐ Sun

Blackout times apply to all selected targets.
The selected targets will not be checked during the indicated blackout periods.

Accept Cancel

Use the AvailabilityMonitor (Set Blackout Times) dialog box to specify the blackout properties for the selected targets. The blackout properties are described in Table 2-13, “Blackout Properties,” on page 2-56.

Note

The availability blackout options do not include the option to merge the blackout periods.

Step 4 Define the blackout properties, and click **Accept**.

Setting Notification Targets

Summary: Set up specific targets for PATROL KM for Event Management notifications to ensure that the proper people are notified of alert conditions. In this task, you set the notification target for a parameter target.

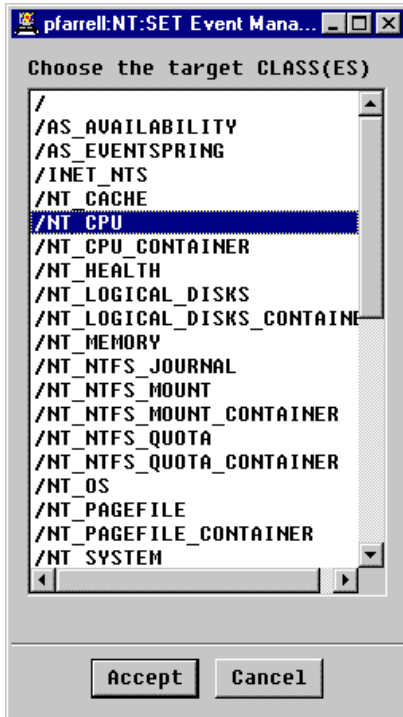
Setting Notification Targets for a Parameter

To set the notification target for a parameter perform the following steps:

- Step 1** Right-click the computer icon of a managed system.
- Step 2** Choose **KM Commands => Event Management => Alert Settings => Notification Targets => Email => Local Targets ANY STATUS => Set For Parameters**.

The Set Event Management KM Variables/Rules (Choose the Target Classes) dialog box opens as shown in Figure 2-18 on page 2-60.

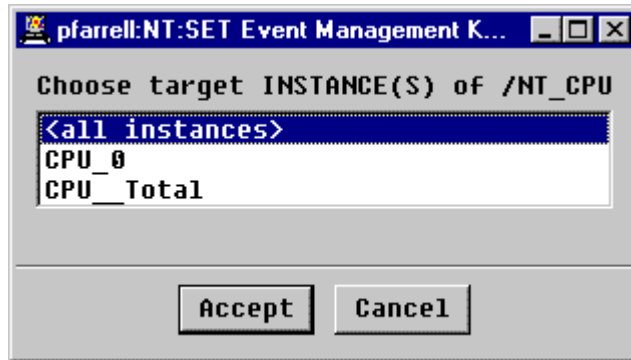
Figure 2-18 Set Event Management KM Variables/Rules (Choose the Target Classes) Dialog Box



Step 3 Select the application class of the parameter, and click **Accept**.

The Set Event Management KM Variables/Rules (Choose target Instances) dialog box opens as shown in Figure 2-19.

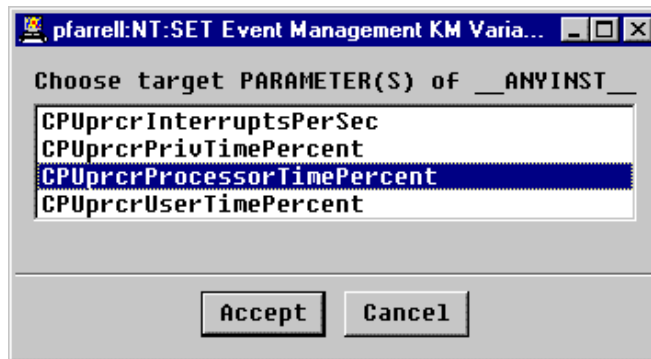
Figure 2-19 Set Event Management KM Variables/Rules (Choose target Instances) Dialog Box



Step 4 Select the application instance of the parameter, and click **Accept**.

The Set Event Management KM Variables/Rules (Choose target Parameters) dialog box opens as shown in Figure 2-20.

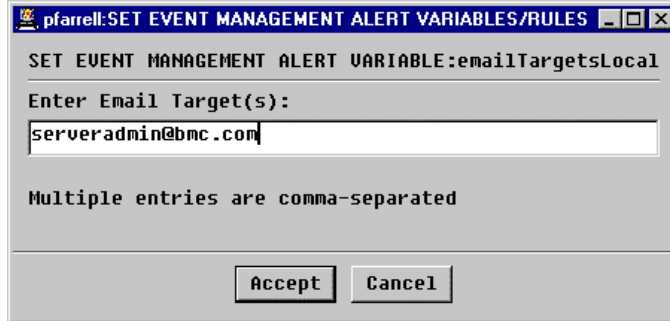
Figure 2-20 Set Event Management KM Variables/Rules (Choose target Parameters) Dialog Box



Step 5 Select the parameter, and click **Accept**.

The Set Event Management KM Variables/Rules (Set Event Management Alert Variables) dialog box opens as shown in Figure 2-21.

Figure 2-21 Set Event Management Alert Variables/Rules (Set Event Management Alert Variables) Dialog Box



Step 6 Enter the e-mail address of the target for this alert, and click **Accept**.

You can set other notification targets using this task, but choose the menu command in Step 2 that applies to your target type.

PATROL Objects, Configuration Variables, and Event Management Rules

This chapter provides information on PATROL objects, configuration variables, and event management rules. You can use these objects, variables, and rules with PATROL KM for Event Management to manage events in your PATROL environment.

PATROL Objects	3-2
Example: PATROL Object	3-3
Example: Rule Inheritance	3-4
Example: PATROL KM for Event Management E-mail Rule . . .	3-5

PATROL Objects

Before you use PATROL KM for Event Management, you should have understand PATROL objects. Table 3-1 lists the hierarchy and description of PATROL objects.

Table 3-1 PATROL Objects Hierarchy

Hierarchy (1=Highest)	Object	Description	Windows NT Example	Unix Example
1	host	host computer of the objects that are being monitored	ntprod1	sunprod1
2	application class	type of objects that can be monitored	NT_LOGICAL_DISKS	FILESYSTEM
3	application instance	actual instance of an application class	C:	root
4	parameter	specific parameter of an application instance	LDIdFreeMegabytes	FSCapacity

Example: PATROL Object

This example assumes an NT Server with a physical disk partitioned into multiple logical disks (C: and D:). The logical disk application class contains the parameter, `LDldFreeMegabytes`. This parameter shows the number of megabytes of free space available on the logical drive.

Table 3-2 Example: PATROL Object

Object	Description
host	The NT Server is the host and is represented as a slash: /.
application class:	Logical disks is an application class of the NT Server and is represented as <code>/NT_LOGICAL_DISKS</code> .
application instance	Each named logical disk is an application instance. The logical disk, C:, is an instance of the application class, <code>NT_LOGICAL_DISKS</code> , and is represented as <code>/NT_LOGICAL_DISKS/C:</code> .
parameter	The parameter for application instance C: is represented as <code>/NT_LOGICAL_DISKS/C:/LDldFreeMegabytes</code> .

Example: Rule Inheritance

You can use PATROL KM for Event Management to define different rules for each PATROL object. The KM applies inheritance to these rules. Rules defined for objects at a higher level in the hierarchy are applied to all lower-level objects that do not have their own rule.

This example contains the following rules:

- Rule 1:

E-mail User1 for any parameter in alarm on this host ('/') that does not have a more specific rule assigned. User1 is the default e-mail account.
- Rule 2:

E-mail User2 for any parameter in any instance of the logical disks application class (/NT_LOGICAL_DISKS) in alarm on this host.
- Rule 3:

E-mail User3 for any parameter in alarm on this host for the logical disks application class instance C: (/NT_LOGICAL_DISKS/C:).
- Rule 4:

E-mail User4 when the parameter, LDldFreeMegabytesfor, is in alarm on this host for the logical disks application class instance C: (/NT_LOGICAL_DISKS/C:/LDldFreeMegabytes).

Table 3-3 Example: Rule Inheritance

Object in Alarm	Action
/NT_LOGICAL_DISKS/C:/LDldFreeMegabytes	User4 receives an e-mail. Rule 4 is specific to the parameter and instance that are in alarm
/NT_LOGICAL_DISKS/C:/LDldIdleTimePercent	User3 receives e-mail. No rule is defined for the parameter, LDldIdleTimePercent, so the parameter inherits the rules defined for the application instance, (logical disk C:), which, in this case, is Rule 3.
/NT_LOGICAL_DISKS/D:/LDldFreeMegabytes	User2 receives an e-mail. No rule is defined at the instance level for D:, so Rule 2, the application class rule, applies.
/NT_CPU/CPU_0/CPUprcrProcessorTimePercent	User1 receives an e-mail since none of the lower-level rules apply to this alarm.

Example: PATROL KM for Event Management E-mail Rule

PATROL KM for Event Management uses variables to specify which functions are performed when events occur for a PATROL object. A variable is defined by a category, a PATROL object, and a rule.

The general variable naming format is the following:

<category>/<object>/<rule>

Table 3-4 PATROL KM for Event Management Variable Definition

<category>	a grouping of related rules that perform notification
<object>	the PATROL object to which the rule applies If <object> is not specified, the rule applies to all objects.
<rule>	the rule to apply to the PATROL object

The following example shows how to send an e-mail when an alarm condition occurs for a PATROL object:

```
/AS/EVENTSPRING/ALERT/EMAIL/NT_LOGICAL_DISKS/C:/emailTargetsLocalALARM
```

<category>: /AS/EVENTSPRING/ALERT/EMAIL is the category including rules that perform notification.

<object>: /NT_LOGICAL_DISKS/C: is the PATROL object, an application instance, to which the rule is applied.

<rule>: emailTargetsLocalALARM is the rule indicating who is notified by e-mail when an alarm condition occurs for the PATROL object
/NT_LOGICAL_DISKS/C:

PATROL KM for Event Management uses a special instance name of __ANYINST__ to indicate that a rule for a particular parameter is applied to all instances of the application class. For example,

```
/AS/EVENTSPRING/ALERT/EMAIL/NT_SERVICES/__ANYINST__/ServiceStatus/emailTargetsLocal
```

causes an e-mail to be sent if an alarm occurs for the **ServiceStatus** parameter on any monitored service on the NT Server.

The PATROL object name can also include the hostname when the rule exists on the NS but only applies to a particular host. For example, for the configuration variable

```
/AS/EVENTSPRING/ALERT/EMAIL/ntprod1/NT_LOGICAL_DISKS/C:/emailTargetsLocalALARM
```

the rule only applies for alarm conditions occurring on the NT Server, ntprod1.

PATROL KM for Event Management provides menu commands to create the rules that are used for managing your PATROL environment eliminating or minimizing manual editing of the variables.

Menu Commands

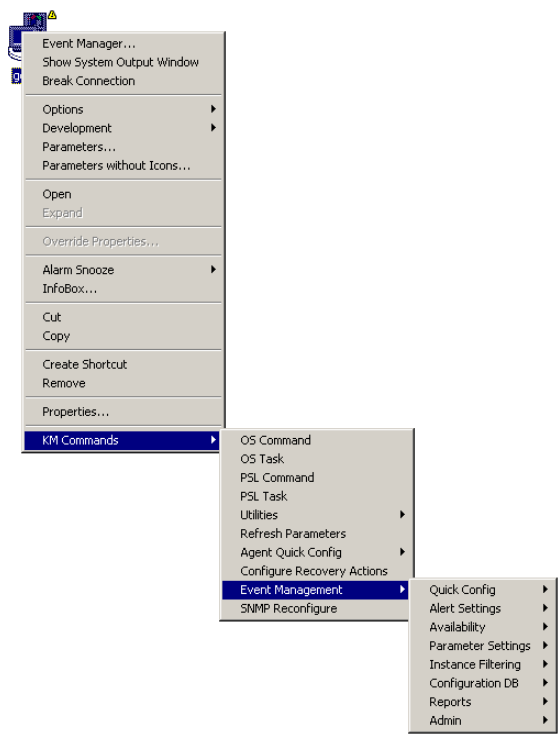
This section describes PATROL KM for Event Management menu commands. You can use the menu commands to perform configuration and reporting tasks and to set up availability monitoring. You access the menu commands from the computer icon and EVENT_MANAGEMENT application icon.

COMPUTER Menu Commands	4-2
Quick Config	4-2
Alert Settings	4-3
Availability	4-18
Parameter Settings	4-22
Instance Filtering	4-24
Configuration DB	4-25
Reports	4-25
AS_EVENTSPRING Application Menu Commands	4-28
About	4-28
Manage Events	4-28
Reports	4-29
Alert Testing	4-29
Refresh Parameters	4-29
AS_AVAILABILITY Application Menu Commands	4-30
About	4-30
Alert Testing	4-30
Refresh Parameters	4-31

COMPUTER Menu Commands

You can use PATROL KM for Event Management menu commands (Figure 4-1) to configure alert settings, availability monitoring, and parameter settings.

Figure 4-1 Computer Menu Commands



Quick Config

You can use the Quick Config menu commands to quickly set up notification servers and remote agents.

Alert Settings

You can use the Alert Settings menu commands to set up notification and recovery actions. Each PATROL object (computer, application class, instance, and parameter) can have its own alert settings.

Alert Actions

You can use the Alert Actions menu commands to specify the actions to perform when an alert condition occurs for a specific PATROL object.

Table 4-1 Alert Actions

Alert Actions	Description
Allow Recovery Actions	Enables recovery procedures (typically a script) to run when an alert condition occurs. The recovery procedure must be specified.
Trigger NOTIFY_EVENT (Perform Notification)	Indicates a NOTIFY_EVENT triggers and that notification of some type should be sent. See also "Notification Command" on page 4-6 and "Remote Alert Settings" on page 4-8.

Table 4-2 Alert Actions Configuration

Rule	arsAction
Configuration Variable	/AS/EVENTSPRING/<object>/arsAction
Values	3 bit mask 0 - Do nothing 1 - Testing 2 - Allow recovery actions 3 - Testing and allow recovery actions 4 - Trigger NOTIFY_EVENT (Perform notification) 5 - Notification and testing 6 - Allow recovery and notification 7 - Notification, testing, and allow recovery

Notification System

You can use the Notification System menu commands (Table 4-3) to specify the location of the notification server that will perform the notification for the specified PATROL object. The notification server can be local, remote, both, or none.

Table 4-3 Notification System

Location	Description
Local	<p>Send alerts to a notification server on the same computer as the PATROL Agent that is doing the monitoring. This configuration is sufficient for sending e-mail notification, but most systems do not have paging capabilities. Paging usually requires the Remote option.</p> <p>Using the Local option, notification failures have no impact on other systems. Local notification is potentially more reliable since it requires fewer intermediate components (for example, the network connection, the remote notification server).</p> <p>Some disadvantages to using the Local option are</p> <ul style="list-style-type: none">• Notification scripts and procedures are required on each computer.• Most systems are unable to perform paging locally.• Notification targets (for example, who is paged or e-mailed) must be maintained for each computer.
Remote	<p>Send alerts to a notification server on a different computer. You can specify a primary and a backup notification server. If the primary notification server is unavailable, fail-over to the backup notification server is automatically performed.</p> <p>Using Remote, allows for centralized notification and simplifies maintenance of notification settings and procedures. A disadvantage is that the failure of a notification server affects many systems.</p>
None	PATROL KM for Event Management does not perform notification.

Table 4-4 Notification System Configuration

Rule	alertSystem
Configuration Variable	/AS/EVENTSPRING/ALERT/<object>/alertSystem
Values	Remote Local Local, Remote None

Local Alert Settings

You can use the Local Alert Settings menu commands to specify configurations or rules that apply to the PATROL Agent where the alert occurs. Local settings can include external procedures (for example, script, batch file, or other OS command) to execute for local notification and recovery actions. Local notification settings (for example, locally defined targets and the notification command) are used only if the value of the Notification System is set to Local. If the Notification System is set to Remote, local alert settings, including local notification targets, are not forwarded to the notification server. The Local Alert Settings commands are

- alertResend
- alertLocalCommand
- arsCommand
- arsCmdType
- alertResendOnInit

Alert Resend

You can use the Alert Resend menu command to specify the number of times the agent should resend outstanding alerts. You can configure alarm and warning alerts to have different resend values. Alerts are resent at the interval specified by the ResendAlertQueue parameter polling time and will contain current alert information, such as, parameter value and status. A resend value of -1 causes PATROL KM for Event Management to resend outstanding alerts for <object> until the alert condition clears.

Table 4-5 Local Alert Settings: Alert Resend Configuration

Rule	alertResend
Configuration Variable	/AS/EVENTSPRING/ALERT/<object>/alertResend
Values	<Alarm_Resends>,<Warning_Resends>
Example	2,1

Notification Command

You can use the Notification Command menu command to specify a script or program to perform notification, such as paging or e-mail. PATROL KM for Event Management contains sample notification scripts located in the PATROL **PSL** directory. The same notification command is typically set at the root (/) object level so that it applies to all PATROL objects (for example, application classes, instances, and parameters) with an ALERT status.

Table 4-6 Local Alert Settings: Notification Command Configuration

Rule	alertLocalCommand
Configuration Variable	/AS/EVENTSPRING/ALERT/<object>/alertLocalCommand
Values	<notification_script_or_program>
Example	/usr/patrol/my_notify.sh

Recovery Action Command

You can use the Recovery Action Command menu command to specify a script or program that performs recovery procedures pertaining to the PATROL object with an ALERT status. PATROL KM for Event Management contains sample notification scripts located in the PATROL **PSL** directory.

Table 4-7 Local Alert Settings: Recovery Action Command Configuration

Rule	arsCommand
Configuration Variable	/AS/EVENTSPRING/ALERT/<object>/arsCommand
Values	<recovery_script_or_program>
Example	/usr/patrol/filesystem_recovery.sh

Note

You must enable recovery actions using the Alert Actions menu command for the KM to execute these actions. See “Alert Actions” on page 4-3.

Recovery Action Command Type

You can use the Recovery Action Command Type menu command to specify the command type to use when executing the recovery action command. This command is typically used when the recovery action requires special KM information, such as the PATROL password used to log into a database. The AS_EVENTSPRING application class contains a sample PATROL command type.

Table 4-8 Local Alert Settings: Recovery Action Command Type Configuration

Rule	arsCmdType
Configuration Variable	/AS/EVENTSPRING/ALERT/<object>/arsCmdType
Values	<command_type>
Example	RA_CMD

Send Reset On Init

You can use the Send Reset On Init menu command to specify whether the agent should send, upon an agent restart, a Reset alert for each outstanding alert condition (for example, alarm or warning) existing prior to the agent being shutdown. This rule does not apply to particular objects. By default, this option is not activated.

Table 4-9 Local Alert Settings: Send Reset On Init Configuration

Rule	alertResendOnInit
Configuration Variable	/AS/EVENTSPRING/ALERT/alertResendOnInit
Values	0 - Do not send reset alert 1 - Send reset alert

Remote Alert Settings

You can use the Remote Alert Settings menu commands to specify the notification servers and the communication settings you want to use for remote notifications. The notification server must be running when you set up PATROL KM for Event Management to verify communication between the notification server and the agents that are being configured.

Configure Notification Servers

You can use the Configure Notification Servers menu command to identify the primary and backup notification servers. You use the PATROL Agent connection settings to identify the notification servers.

Table 4-10 Remote Alert Settings: Configure Notification Servers Configuration

Configuration Variable	/AS/EVENTSPRING/NOTIFICATION_SERVER1 (primary) /AS/EVENTSPRING/NOTIFICATION_SERVER2 (backup)
Values	<host>,<agent_port_no>,<user>,<encrypted_password>
Example	ns1,3181,patrolns,FB0A195D062696

Remote Comm Settings

You can use the Remote Comm Settings menu command to configure the remote communication settings to use when sending events to a notification server and PATROL Agent Availability Checking.

Table 4-11 Remote Alert Settings: Remote Comm Settings Configuration

Configuration Variable	/AS/EVENTSPRING/RemoteAgentCommSettings
Values	TCP UDP,<timeout>,<retries>
Example	UDP,100,3

Notification Targets

You can use the Notification Targets menu command to specify who to notify (for example, paged or e-mailed) when an alert condition occurs. You can specify a different target for WARNING, ALARM, INFORMATION, and ESCALATED conditions. You can also generically specify notification targets for any alert status when the target is the same for all alert conditions.

Each notification server has its own set of targets. Remote notification targets are forwarded to the notification server for processing and are only defined if the notification system is set to Remote. Notification targets are typically only defined on the notification server.

None Target

You can use a special target name of None to prevent targets from being inherited or to prevent notification. For example, if you set a default pager target for an application class, setting the paging target to None for a particular instance or parameter in that application class will prevent paging.

Note

The target type, such as page or e-mail, can be changed to any type of notification action. For example, trouble ticket targets are not required for trouble tickets. The target can perform any task programmed in the notification command.

Email Target

You can use the Email target to specify e-mail accounts as notification targets.

Table 4-12 Notification Targets: Email Target Configuration

Rule:	emailTargets
Configuration Variable	/AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsLocalINFORMATION /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsLocalWARNING /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsLocalALARM /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsLocalESCALATED /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsLocal /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsRemoteINFORMATION /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsRemoteWARNING /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsRemoteALARM /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsRemoteESCALATED /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsRemote
Values	<email_target1>,<email_target2>,...<email_targetn>
Example	patrol,admin@company.com

Pager Target

You can use the Pager Target to specify one or more pagers as notification targets.

Table 4-13 Notification Targets: Pager Target Configuration

Rule	pagerTargets
Configuration Variable	/AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsLocalINFORMATION /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsLocalWARNING /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsLocalALARM /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsLocalESCALATED /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsLocal /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsRemoteINFORMATION /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsRemoteWARNING /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsRemoteALARM /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsRemoteESCALATED /AS/EVENTSPRING/ALERT/PAGER/<object>/pagerTargetsRemote
Values	<pager_target1>,<pager_target2>,...<pager_targetn>
Example	adminpg

Custom Target

You can use the Custom Target to specify custom notification targets.

Table 4-14 Notification Targets: Custom Target Configuration

Rule	customTargets
Configuration Variable	/AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsLocalINFORMATION /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsLocalWARNING /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsLocalALARM /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsLocalESCALATED /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsLocal /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsRemoteINFORMATION /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsRemoteWARNING /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsRemoteALARM /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsRemoteESCALATED /AS/EVENTSPRING/ALERT/CUSTOM/<object>/customTargetsRemote
Values	<custom_target1>,<custom_target2>,...<custom_targetn>
Example	glamis

TT Target

You can use the TT Target to specify TT notification targets.

Table 4-15 Notification Targets: TT Target Configuration

Rule	ttTargets
Configuration Variable	/AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsLocalINFORMATION /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsLocalWARNING /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsLocalALARM /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsLocalESCALATED /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsLocal /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsRemoteINFORMATION /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsRemoteWARNING /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsRemoteALARM /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsRemoteESCALATED /AS/EVENTSPRING/ALERT/TT/<object>/ttTargetsRemote
Values	<trouble_tix_target1>,<trouble_tix_target2>,...<trouble_tix_targetn>
Example	ttpatrol

Note

If the type of alert condition (INFORMATION, WARNING, ALARM, or ESCALATED) is not specified for a notification target (for example, /AS/EVENTSPRING/ALERT/EMAIL/<object>/emailTargetsRemote), notification occurs for all alert conditions.

Alert Messages

You can use the Alert Messages menu commands to customize the default notification message format or define a unique message format for specific parameters, application classes, or instances. The notification message is displayed on e-mails, pages, and events.

Table 4-16 Alert Messages Configuration

Rule	msgText
Configuration Variable	/AS/EVENTSPRING/ALERT/MSG/<object>/msgText /AS/EVENTSPRING/ALERT/MSG/<object>/msgTextINFORMATION /AS/EVENTSPRING/ALERT/MSG/<object>/msgTextWARNING /AS/EVENTSPRING/ALERT/MSG/<object>/msgTextALARM
Values	<message_replacement_text> <message_replacement_variables> <message_replacement_text_and_variables>
Example	%HOSTNAME% has CPU Processor time of %PARAMETER_VALUE% for time zone %TIMEZONE% for %OS_TYPE%

Note

If you do not specify INFORMATION, WARNING or ALARM, the same message format is used for all events.

Table 4-17 Alert Messages Replacement Variables

Replacement Variable	Description	Example
%HOSTNAME%	hostname of the computer where the alert occurred	glamis.agentspring.com
%IPADDRESS%	IP Address of %HOSTNAME%	192.168.1.1
%TCP_PORT%	PATROL Agent TCP port on %HOSTNAME%	3181
%UDP_PORT%	PATROL Agent's UDP port on %HOSTNAME%	3181
%APPCLASS%	application class name	FILESYSTEM
%APPINSTANCE%	instance name (sid) internal to the agent	root

Table 4-17 Alert Messages Replacement Variables

%ICON_NAME%	name of the instance as it appears on the PATROL Console	root
%PARENT_INSTANCE%	instance name of the parent container of %APPINSTANCE%	/ORACLE/ORACLE
%PARAMETER_NAME%	parameter name	FSCapacity
%PARAMETER_VALUE%	parameter value	99.00
%PARAMETER_STATUS%	parameter status	ALARM
%DATE%	date the alert occurred on %HOSTNAME% Format: MM/DD/YYYY	03/30/2001
%TIME%	time the alert occurred on %HOSTNAME% Format: HH:MM:SS	06:26:21
%TIMEZONE%	time zone on %HOSTNAME%	US/Eastern/EDT
%LAST10%	last 10 parameter values (space delimited)	95.43 97.12 98.00 99.54 98.01 ...
%AVE10%	average of %LAST10%	97.34
%LAST10TS%	last 10 parameter timestamps of parameter values listed in %LAST10%	957359389 957359689 957359989 ...
%LAST10TP%	overall time period, in minutes, of %LAST10%	50.00
%USERDEFINED%	special user-defined data passed to PATROL KM for Event Management	
%OS_TYPE%	operating system	SOLARIS 5.8
%ALARM_MIN%	lower threshold of current alarm range	90
%ALARM_MAX%	upper threshold of current alarm range	100

Table 4-17 Alert Messages Replacement Variables

%CUSTOM_ID1%	custom identifier assigned to alarming object For example, the application, prod_app1, is tied to the object, filesys1. When filesys1 goes into alarm, a message can be displayed indicating that filesys1 is almost full, which affects the production application1, prod_app1.	prod_app1
%CUSTOM_ID2%	custom identifier assigned to alarming object	department1
%EVENT_ID%	event id for the alert	9875
%EVENT_TYPE%	event type for the alert	ALARM
%EVENT_STATUS%	event status for the alert	OPEN
%NOTIFY_EVENT_ID%	event id for the generated NOTIFY_EVENT Note: Variable is only available from a notification server.	5439
%NOTIFY_EVENT_TYPE%	event type for the generated NOTIFY_EVENT Note: Variable is only available from a notification server.	ALARM
%NOTIFY_EVENT_STATUS%	event status for the generated NOTIFY_EVENT Note: Variable is only available from a notification server.	OPEN
%AGENT_VERSION%	PATROL Agent version Note: Variable is only available with PATROL KM for Event Management 2.5.00 or later.	3.5.00

Table 4-17 Alert Messages Replacement Variables

%EVENT_CATALOG%	event catalog name for the originating alert Note: Variable is only available with PATROL KM for Event Management 2.5.00 or later	0
%EVENT_CLASS%	event class name for the originating alert Note: Variable is only available with PATROL KM for Event Management 2.5.00 or later.	11
%EVENT_SEVERITY%	event severity for the originating alert Note: Variable is only available with PATROL KM for Event Management 2.5.00 or later.	4

An example of a reworded message template is:

```
%HOSTNAME% has CPU Processor time of %PARAMETER_VALUE% for  
time zone %TIMEZONE% for %OS_TYPE%
```

At run time, the message could be displayed as:

```
Mercury has CPU Processor time of 99 for time zone Eastern  
Standard Time for NT 5.0 Service Pack 1
```

Note

A special instance name of `__ANYINST__` can be used for all instances of a parameter. If `<object>` is not specified, then the configuration variable defines the default message format.

Blackout Periods

You can use the Blackout Periods menu commands to prevent notification during a specified time period, even if an alert condition occurs. You can specify multiple blackout times per day. Blackout periods currently apply to notification only and can be applied to most PATROL objects.

Blackout periods are similar to notification targets: they can be defined locally at the system where the alert occurs and at the notification server. If defined locally and Notification System is set to Remote, alerts are not forwarded from the originating system to the notification server during a defined blackout period. If the blackout period is defined at the notification server, alerts are forwarded by the originating systems to the notification server during defined blackout periods, but notification is not performed even though an alert is received. The notification is stopped by the notification server.

Table 4-18 Blackout Periods Configuration

Rule	blackoutPeriod
Configuration Variable	/AS/EVENTSPRING/BLACKOUT/<object>/blackoutPeriod
Values	<day1> <start1> <stop1>,<day2> <start2> <stop2> Note: Start and stop times are in seconds past midnight.
Example	Sat 3600 7200,Wed 3600 7200

Notification Server Settings

You can use the Notification Server Settings menu command to perform actions specific to a notification server.

Remote Target Setting Menu Command

You can use the Remote Target Setting menu command to specify the handling of remote targets received by the notification server. Table 4-19 lists the options you can specify for the remote target setting. The default setting for this rule is Merge.

Table 4-19 Remote Target Setting Options

Option	Description
Merge with Local	Remote targets are merged with local targets defined at the notification server (default).
Override Local	Only remote targets are used for notification.
Ignore Remote Targets	Remote targets are ignored.

Table 4-20 Notification Server Settings: Remote Target Setting Configuration

Rule	nsRemoteTargetSetting
Configuration Variable	/AS/EVENTSPRING/NS/nsRemoteTargetSetting
Values:	Ignore Merge (default) Override
Example	Merge

Custom Identifiers

You can use the Custom Identifiers menu commands (Custom Id1 and Custom Id2) to assign custom identifiers to an object.

Table 4-21 Custom Identifiers Configuration

Configuration Variable	/AS/EVENTSPRING/<object>/customId1 /AS/EVENTSPRING/<object>/customId2
Values	<custom_identifier>
Example	financial_app

Overrides

You can use the Overrides menu command to override PATROL KM for Event Management rules based on the time of day. Using this command, you can establish multiple overrides per day.

Table 4-22 Overrides Configuration

Configuration Variable	/AS/EVENTSPRING/_/_OVERRIDE_/_/<object>/<rule>
Values	<day> <start> <stop>=<value_for_rule>
Example	Sat 0 86399,Sun 0 86399=oncallpager

Availability

You can use the Availability menu commands to specify the agents and hosts that PATROL should monitor for availability.

Add Target

You can use the Add Target menu command to add additional agents and hosts to be monitored. If a PATROL Agent port is specified, the KM performs PATROL Agent Availability Checking. SNMP monitoring is similar. You can also use this menu command to change the default SNMP settings and the SNMP Object ID.

Table 4-23 Add Target Configuration

Configuration Variable	/AS/EVENTSPRING/AVAILABILITY/Targets /AS/EVENTSPRING/AVAILABILITY/Targets2
Values	<host> <PATROL_Agent_port> <SNMP_port>
Example	glamis 3181 161,mirage - 161

Configuration Variable	/AS/EVENTSPRING/AVAILABILITY/snmpSettings
Values	<SNMP_Community> <SNMP_Timeout> <SNMP_Retries>
Example	public,500,3

Configuration Variable	/AS/EVENTSPRING/AVAILABILITY/snmpOid
Values	<SNMP_ObjectID>
Example	.1.3.6.1.2.1.1.3.0

Updated Flag

You must set the Updated configuration variable to 1 before PATROL KM for Event Management recognizes changes to the Targets and Primary configuration variables.

Table 4-24 Add Target: Updated Flag Configuration

Configuration Variable	/AS/EVENTSPRING/AVAILABILITY/Updated
Values	0 - Ignore changes to Targets and Primary configuration variables 1 - Enable changes to Targets and Primary configuration variables

Note

The KM automatically sets the Updated variable to 1 whenever you make a change using the Availability menu command.

Remove Targets

You can use the Remove Targets menu command to remove agents and hosts from the monitoring list that were added using the Add Target menu command.

Failover Settings

You can use the Failover Settings menu commands allow you to specify or remove a primary availability monitor.

Identify Primary

You can use the Identify Primary menu command to select the primary monitor from the list of configured availability targets. The primary monitor must be added to the target list before running this command. If the primary is unavailable, all configured availability targets are monitored. Once the primary becomes available, it becomes the only monitored target.

Note

The PATROL Agent that you use to run the Identify Primary menu command becomes the backup monitor.

Table 4-25 Failover Settings: Identify Primary Configuration

Configuration Variable	/AS/EVENTSPRING/AVAILABILITY/Primary
Values	<host_primary_monitor> <PATROL_Agent_port> <SNMP_port>
Example	pismo 3181 161

Remove Primary

You can use the Remove Primary menu command to remove the primary monitor you specified using the Identify Primary menu command. After executing the Remove Primary menu command, all configured availability targets are monitored.

Blackout Periods

You can use the Blackout Periods menu command to specify time periods to stop monitoring of a host or an agent. To prevent unwanted alerts, you can specify a blackout period for all computers that have scheduled reboots. You can specify multiple availability blackout periods per day.

Note

If you specify one or more availability blackouts for a PATROL Agent or host, the KM stops monitoring that agent or host during the blackout period.

Table 4-26 Availability: Blackout Periods Configuration

Configuration Variable	/AS/EVENTSPRING/AVAILABILITY/Blackouts/<host_port>
Values	<day1> <start1> <stop1>, <day2> <start2> <stop2>
Example	Sat 3600 7200,Wed 3600 7200

Report Targets

You can use the Report Targets menu command to display information about the hosts and agents being monitored and any blackout periods defined for the monitored hosts and agents.

Ping Command

You can use the Ping Command menu command to specify the operating system command to use when checking host availability. This setting overrides the default ping command used by PATROL KM for Event Management.

Table 4-27 Availability: Ping Command Configuration

Configuration Variable	/AS/EVENTSPRING/AVAILABILITY/pingCmd
Values	<absolute_path_to_OS_ping_command>
Example	/usr/patrol/myping.sh

Checker Account

You can use the Checker Account menu command to configure the checker agent to use a user-defined account when performing pings against other PATROL Agents.

Table 4-28 Availability: Checker Account Configuration

Configuration Variable	/AS/EVENTSPRING/AVAILABILITY/checkerAccount
Values	<username>,<PEM_encrypted_password>
Example	patrolns,FB0A195D062696

Note

BMC Software recommends that you do use the PATROL Agent default account. The account could be locked out if an agent check fails because of an invalid login.

Parameter Settings

You can use the Parameter Settings menu commands to customize PATROL parameter thresholds and polling times. These changes are stored externally from the KM and do not change the KM version.

Thresholds

The Thresholds menu command enables to change the thresholds for all instances of a parameter or thresholds for a particular parameter instance. Changes apply only to the agent where the command was executed. This operation is similar to creating a localized parameter (overriding the global definition) by using a developer console.

Table 4-29 Parameter Settings: Thresholds Configuration

Configuration Variable	/AS/EVENTSPRING/PARAM_SETTINGS/THRESHOLDS/<parameter> where <parameter> is the full parameter object name such as /FILESYSTEM/root/FSCapacity
Value	<parameter_settings>
Example	1,1 0 100 0 0 2,1 1 50 0 0 1,1 51 100 0 0 2

Polltimes

You can use the **Polltimes** menu command to specify poll times in minutes for standard and collector parameters. The KM converts the poll time values you specified to seconds before it stores them. For example, 1.5 minutes is converted to 90 seconds.

Table 4-30 Parameter Settings: Polltimes Configuration

Configuration Variable	/AS/EVENTSPRING/PARAM_SETTINGS/POLLTIMES/<parameter>/interval where <parameter> is the full parameter object name such as /FILESYSTEM/root/FSCapacity
Values	<pooltime_in_seconds>
Example	90

Status Flags

You can use the Status Flags menu command to enable or disable the use of PATROL KM for Event Management thresholds or poll times.

Table 4-31 Parameter Settings: Status Flags Configuration

Configuration Variable	/AS/EVENTSPRING/PARAM_SETTINGS/STATUSFLAGS/<parameter>/paramSettingsStatusFlag where <parameter> is the full parameter object name such as /FILESYSTEM/root/FSCapacity
Values	0 - PATROL KM for Event Management settings will be ignored 1 - PATROL KM for Event Management parameter settings have been processed 2 - Refresh thresholds 4 - Refresh poll times 6 - Refresh thresholds and poll times

Note

You can use a special instance name, __ANYINST__, for all instances of a parameter.

Instance Filtering

You can use the Instance Filtering menu commands to manage discovered application instances.

Edit Filter List

You can use the Edit Filter List menu command to select the application instances you want to filter.

Table 4-32 Instance Filtering: Edit Filter List Configuration

Configuration Variable	/AgentSetup/<ApplicationClass>.filterList
Values	<list_of_application_instances>
Example	C:,D:,E:,F:

Change Filter Type

You can use the Change Filter Type menu command to specify the filter type that is applied to the application instances in the filter list. There are two filter types: Exclude and Include.

Exclude

You can use the Exclude filter type to stop monitoring all instances specified in the filter list.

Include

You can use the Include filter type to specify that only those instances specified in the filter list are monitored.

Table 4-33 Instance Filtering: Edit Filter List Configuration

Configuration Variable	/AgentSetup/<ApplicationClass>.filterType
Values	Include Exclude

Filtered Instance Report

You can use the Filtered Instance Report menu command to generate a report listing the filter type and filtered instances for each application class.

Configuration DB

You can use the Configuration DB menu commands to view and delete configuration variables for PATROL KM for Event Management.

Display Values

You can use the Display Values menu command to display the values stored in the configuration database for selected configuration variables.

Delete Variables

You can use the Delete Variables menu command to delete selected configuration variables from the configuration database.

Reports

You can use the Reports menu commands to display reports of any recovery action output and the current parameter settings.

Parameter Settings

You can use the Parameter Settings menu command to create a report of active parameters in the KM. Selecting the Parameter Settings menu command displays the PARAMETER SETTINGS REPORT dialog box (Figure 4-2).

Figure 4-2 PARAMETER SETTINGS REPORT Dialog Box

geminiParameter Report Event Management VARIABLES

PARAMETER SETTINGS REPORT

Output Type:
☒ Report ☐ CSV (Comma Separated Values) ☐ Names Only

Object Name Filter:

Property Filters:

- ☐ Return parameters matching ALL filter selections
- ☐ Return parameters with active thresholds
- ☐ Return consumer parameters
- ☐ Return Standard/Collector parameters
- ☐ Return parameters with Recovery Actions
- ☐ Return parameters in ALARM
- ☐ Return parameters in WARNING
- ☐ Return parameters localized in KM

Accept **Cancel**

The report formats are listed in Table 4-34.

Table 4-34 Parameter Settings Report Formats

Report Format	Description
Report	a full, detailed report of parameter settings and current parameter values
CSV	a report of parameter settings and current parameter values as a comma-separated list of values
Names Only	a report listing only the full parameter object name

You can filter the report by specifying an object name in the Object Name Filter field and selecting one or more parameter attributes listed in the Property Filters section of the PARAMETER SETTINGS REPORT dialog box.

Recovery Action Output

You can use the Recovery Action Output menu command to display a dialog box listing output files created by parameter recovery actions executed by PATROL KM for Event Management. You can select a file for review.

Admin

You can use the Admin menu command to clear the alert queues.

Clear Alert Queues

You can use the Clear Alert Queues menu command to clear the following PATROL KM for Event Management queues:

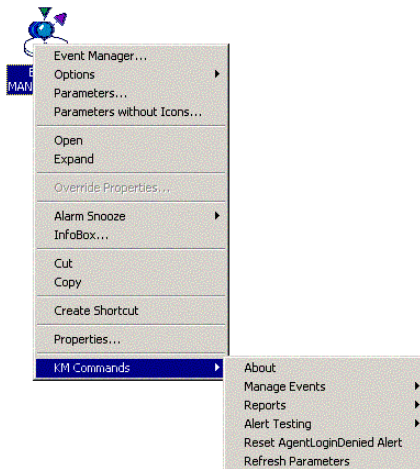
- Resend Alert Queue, which contains outstanding alerts marked to be re-sent to the Notification Server
- Retrigger Event Queue, which contains events that failed to be sent to the Notification Server

You can run this command to stop the processing of queued alerts before you try to resolve any communications problems with the notification server.

AS_EVENTSPRING Application Menu Commands

The menu commands for managing and reporting on events managed by PATROL KM for Event Management are accessed from the AS_EVENTSPRING application icon (Figure 4-3).

Figure 4-3 AS_EventSpring Application Menu Commands



About

You can use the About menu command to display the version of PATROL KM for Event Management currently running on the computer.

Manage Events

You can use the Manage Events menu commands to acknowledge open notification events or close acknowledged notification events.

Acknowledge Open Events

You can use the Acknowledge Open Events menu command to send an acknowledgement to all open PATROL KM for Event Management notification events (NOTIFY_EVENT).

Close Acknowledged Events

You can use the Close Acknowledged Events menu command to close all acknowledged open PATROL KM for Event Management notification events (NOTIFY_EVENT).

Reports

You can use the Reports menu commands to create reports based on one of the following:

- all Open notification events
- all acknowledged notification events
- all escalated notification events
- all notification events

Alert Testing

You can use the Alert Testing menu commands to force the AlertTest parameter into an Alarm (Test ALARM) or Warning (Test WARN) condition. This command is primarily used for testing the notification settings and notification servers. If you configure the alert action to perform notification, the menu command generates a TEST_NOTIFY_EVENT.

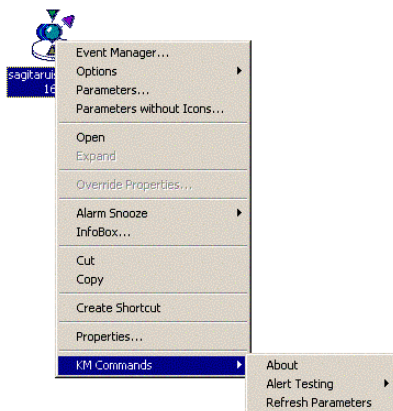
Refresh Parameters

You can use the Refresh Parameters menu command to perform a refresh of all AS_EVENTSPRING application class parameters.

AS_AVAILABILITY Application Menu Commands

The menu commands for managing monitored agents and hosts are accessed from the AS_AVAILABILITY application icon (Figure 4-4).

Figure 4-4 AS_AVAILABILITY Application Menu Commands



About

You can use the About menu command to display the version of PATROL KM for Event Management currently running on the computer.

Alert Testing

You can use the Alert Testing menu commands to force the AlertTest parameter into an Alarm (Test ALARM) or Warning (Test WARN) condition. This command is primarily used for testing the notification settings and notification servers. If you configured the alert action to perform notification, the menu command generates a TEST_NOTIFY_EVENT.

Refresh Parameters

You can use the Refresh Parameters menu command to perform a refresh of all AS_AVAILABILITY application class parameters.

Parameters

This chapter list PATROL KM for Event Management parameters and shows their default settings. The following topics are discussed:

Parameter Summary	5-2
Parameter Defaults	5-3

Parameter Summary

PATROL KM for Event Management parameters are in the following applications classes:

- AS_EVENTSPRING
- AS_AVAILABILITY

Table 5-1 lists PATROL KM for Event Management parameters.

Table 5-1 PATROL KM for Event Management Parameters

Parameter	Application Class	Description
AlertTest	AS_EVENTSPRING AS_AVAILABILITY	used for alert testing
AgentLoginDenied	AS_EVENTSPRING	tracks failed login attempts to the agent
NotifyEvents	AS_EVENTSPRING	tracks NOTIFY events as they are processed
RefreshParamSettings	AS_EVENTSPRING	updates parameter settings as required
ResendAlertQueue	AS_EVENTSPRING	resends specified alerts
RetriggerEventQueue	AS_EVENTSPRING	retriggers events that failed to be sent to a notification server
AvailabilityMonitorColl	AS_AVAILABILITY	collector that performs availability checks
AgentPingFailures	AS_AVAILABILITY	PATROL Agent is down
HostPingFailures	AS_AVAILABILITY	host is unavailable using an ICMP ping
SnmpPingFailures	AS_AVAILABILITY	SNMP Agent is unavailable

Parameter Defaults

Table 5-2 lists the default properties for each PATROL KM for Event Management parameter.

Table 5-2 PATROL KM for Event Management Parameter Defaults

Parameter	Type	Alarm1 Alarm2	Border	Schedule	Icon	Units	Annotated	Active
AlertTest	Con	1-50 51-100	N/A	N/A	Gauge	N/A	N	Y
AgentLoginDenied	Con	N/A N/A	0-0	N/A	Graph	N/A	Y	Y
NotifyEvents	Con	N/A N/A	N/A	N/A	Graph	N/A	Y	Y
RefreshParamSettings	Std	N/A N/A	0-1000	1.5 mins	Graph	N/A	N	Y
ResendAlertQueue	Std	N/A N/A	N/A	10 mins	Graph	N/A	N	Y
RetriggerEventQueue	Std	N/A N/A	N/A	2 mins	Graph	N/A	N	Y
AvailabilityMonitorColl	Coll	N/A N/A	N/A	2 mins	N/A	N/A	N	Y
AgentPingFailures	Con	N/A 1	N/A	N/A	Graph	N/A	N	Y
HostPingFailures	Con	N/A 1	N/A	N/A	Graph	N/A	Y	Y
SnmpPingFailures	Con	1 2	N/A	N/A	Graph	N/A	N	Y

Command-line Interface

PATROL KM for Event Management contains a command-line interface that triggers a **NOTIFY_EVENT** using scripts or batch programs. PATROL KM for Event Management can be configured to initiate notification when the **NOTIFY_EVENT** is triggered. An event origin is specified as an argument to the script and is used as the PATROL KM for Event Management alert object to determine who to notify, etc. Command-line scripts are in the PATROL PSL directory.

Unix: `$PATROL_HOME/lib/psl/AS_PATROL_NOTIFY.sh`

Windows: `%PATROL_HOME%\lib\psl\AS_PATROL_NOTIFY.bat`

See the comments in the script or batch program for more information.

PATROL KM for Event Management Reference

This reference provides a comprehensive listing of PATROL KM for Event Management settings and rules. You can use this reference as a primary source of information for managing monitored objects throughout your enterprise.

PATROL KM for Event Management allows object-level control (for example, application of rules) throughout an enterprise. You can deploy these rules using automation scripting, custom applications, the PATROL Console, or the PATROL Configuration Manager.

Alert and Notification Settings	B-2
Remote Notification Settings	B-8
Notification Server Settings	B-9
Parameter Settings	B-12
Application Class Settings	B-14
Menu Command Access Settings	B-15

Alert and Notification Settings

Table B-1 shows alert and notification settings you can apply to all PATROL objects, including computers, application classes, instances, and parameters. The general variable naming format is:

<CATEGORY> / <OBJECT> / <RULE>

For example, if you want to set the alert action (arsAction) rule specifically for the Unix CPU application class, the variable name is /AS/EVENTSPRING/CPU/arsAction

Table B-1 Alert and Notification Settings

Rule	Description
Category: /AS/EVENTSPRING	
arsAction arsActionINFORMATION arsActionWARNING arsActionALARM Note: arsAction rules based on status are supported in 2.4.05 or later.	a three-bit mask that specifies actions to perform when an alert occurs Values are: 0 = do nothing 1 = reserved 2 = allow recovery actions 4 = perform notification and trigger NOTIFY_EVENT Bit mask values are not mutually exclusive; more than one bit can be set. You can set the arsAction rule in memory to override the persistent configuration setting so that it applies only to the current agent session
arsCommand arsCommandWARNING arsCommandALARM Note: arsCommand rules based on status are supported in 2.4.05 or later.	OS command or PSL script to execute for recovery action procedures Example: /patrol/AS_EVS_RA_CMD.sh
arsCmdType arsCmdTypeINFORMATION arsCmdTypeWARNING arsCmdTypeALARM Note: arsCmdType rules based on status are supported in 2.4.05 or later.	custom command type that you can use with the arsCommand You msut use PSL if arsCommand is PSL code and not a pointer to a PSL script file. Example: RA_CMD

Table B-1 Alert and Notification Settings

Rule	Description
allowOverrides Note: Supported in 2.4.05 or later.	disables the use and lookup of rule overrides (<code>_OVERRIDE_</code>) Values are: 0 = do not allow rule overrides 1 = allow rule overrides (default)
customId1 customId2 Note: Custom identifiers are supported in 2.4.02 or later.	custom identifiers that you can assign to specific PATROL objects Typically used to associate a business application or process with a monitored object. Example: corporate accounting
loginDeniedIgnoredUsers Note: Supported in 2.4.02 or later.	comma delimited list of usernames to ignore when a username is denied login to the agent because of ACL restrictions or invalid login information Example: patrol_checker,EventSpring
useEnvOnlyForCmds Note: Supported in 2.4.02 or later.	indicates whether PATROL KM for Event Management should run all operating system commands (for example, notification and recovery) without command-line arguments PATROL KM for Event Management environment variables are always provided to the operating system command. Values are: 0 = run commands with command-line arguments 1 = do not use command-line arguments
spoolDirectory Note: Supported in 2.4.02 or later.	directory where PATROL KM for Event Management reports are stored (for example, parameter reports and recovery action output) The default location is the directory specified in the <code>PATROL_HOME</code> environment variable. Example: /usr/local/patrol_reports

Table B-1 Alert and Notification Settings

Rule	Description
setParameterValue Note: Supported in 2.5.00 or later.	sets the value of an active parameter Format: PARAMETER=VALUE,DELAY=N,...,PARAMETER=VALUE,DELAY=N where: <ul style="list-style-type: none"> parameter = any valid PATROL parameter object (/APPLICATION_CLASS/INSTANCE/PARM_NAME). When an instance name of _ANYINST_ is used, the KM will select a valid instance. VALUE = ALARM, WARM, OK, CLEAR. If you specify the CLEAR state, the KM will attempt to set the parameter to a value that is not within an active alarm range. DELAY = optional. The time in seconds that the KM will wait between each parameter set. Example: /CPU/CPU/CPUCpuUtil=ALARM, /CPU/CPU/CPUCpuUtil=CLEAR,DELAY=5
Category: /AS/EVENTSPRING/ALERT	
alertLocalCommand	OS command to execute for notification Example: /patrol/AS_EVSLocalAlertNotify.sh
alertSystem alertSystemINFORMATION alertSystemWARNING alertSystemALARM Note: alertSystem rules based on status are supported in 2.4.05 or later.	identifies where notification is performed Values are: <ul style="list-style-type: none"> Local Remote Local, Remote None
alertResend	identifies the number of times to resend alerts Format: Alarm Resends, Warning Resends A value of -1 causes a continuous resend of the alert until the alert condition is cleared. Example: 2,1

Table B-1 Alert and Notification Settings

Rule	Description
alertResetOnInit	<p>indicates whether reset alerts are sent when the agent restarts</p> <p>If enabled, a reset alert is sent for each alarm and warning condition that exists when the agent was last shutdown.</p> <p>Values are: 0 = do not send reset alerts (disable) 1 = send reset alerts (enable)</p>
Category: /AS/EVENTSPRING/ALERT/EMAIL	
emailTargetsLocal emailTargetsLocalINFORMATION emailTargetsLocalWARNING emailTargetsLocalALARM emailTargetsLocalESCALATED emailTargetsRemote emailTargetsRemoteINFORMATION emailTargetsRemoteWARNING emailTargetsRemoteALARM emailTargetsRemoteESCALATED	<p>e-mail targets to be notified when alert condition occurs</p> <ul style="list-style-type: none"> • Multiple targets are comma delimited. • Local targets are passed to the alertLocalCommand. • Remote targets are forwarded to the notification server for processing. • The rules emailTargetsLocal and emailTargetsRemote match any status unless a target was specified for the present alert status. <p>Example: patrol@any.co.com,admin</p>
Category: /AS/EVENTSPRING/ALERT/PAGER	
pagerTargetsLocal pagerTargetsLocalINFORMATION pagerTargetsLocalWARNING pagerTargetsLocalALARM pagerTargetsLocalESCALATED pagerTargetsRemote pagerTargetsRemoteINFORMATION pagerTargetsRemoteWARNING pagerTargetsRemoteALARM pagerTargetsRemoteESCALATED	<p>pager targets to be notified when alert condition occurs</p> <ul style="list-style-type: none"> • Multiple targets are comma delimited. • Local targets are passed to the alertLocalCommand. • Remote targets are forwarded to the notification server for processing. • The rules pagerTargetsLocal and pagerTargetsRemote match any status unless a target was specified for the present alert status. <p>Example: adminpg</p>

Table B-1 Alert and Notification Settings

Rule	Description
Category: /AS/EVENTSPRING/ALERT/CUSTOM	
customTargetsLocal customTargetsLocalINFORMATION customTargetsLocalWARNING customTargetsLocalALARM customTargetsLocalESCALATED customTargetsRemote customTargetsRemoteINFORMATION customTargetsRemoteWARNING customTargetsRemoteALARM customTargetsRemoteESCALATE	target identification for custom notification <ul style="list-style-type: none"> • Multiple targets are comma delimited. • Local targets are passed to the alertLocalCommand. • Remote targets are forwarded to the notification server for processing. • The rules customTargetsLocal and customTargetsRemote match any status unless a target was specified for the present alert status. Example: glamis
Category: /AS/EVENTSPRING/ALERT/TT	
ttTargetsLocal ttTargetsLocalINFORMATION ttTargetsLocalWARNING ttTargetsLocalALARM ttTargetsLocalESCALATED ttTargetsRemote ttTargetsRemoteINFORMATION ttTargetsRemoteWARNING ttTargetsRemoteALARM ttTargetsRemoteESCALATED	targets for generating a trouble ticket when an alert condition occurs <ul style="list-style-type: none"> • Multiple targets are comma delimited. • Local targets are passed to the alertLocalCommand. • Remote targets are forwarded to the notification server for processing. • The rules ttTargetsLocal and ttTargetsRemote match any status unless a target was specified for the present alert status. Example: ttpatrol
Category: /AS/EVENTSPRING/ALERT/MSG	
msgText msgTextINFORMATION msgTextWARNING msgTextALARM	wording for the notification message <p>The message format that you set for the computer object, '/', is used as the default message format</p> Example: ALERT:%PARAMETER_STATUS% %HOSTNAME%. %APPCLASS%.%APPINSTANCE%. %PARAMETER_NAME%=%PARAMETER_VALUE%

Table B-1 Alert and Notification Settings

Rule	Description
Category: /AS/EVENTSPRING/ALERT/BLACKOUT	
blackoutPeriod	<p>comma delimited list of start and stop times when notifications are not sent.</p> <p>The Blackout period format is specified as <day> s1 s2 where s1 and s2 are the number of seconds past midnight.</p> <p>Example: Mon 3600 82800,Tue 3600 82800</p>

Remote Notification Settings

Table B-2 shows alert and notification settings that you can apply to systems configured to perform remote notification. See also Table B-1 on page B-2 for the alert and notification setting, `/AS/EVENTSPRING/ALERT/alertSystem`.

Table B-2 Remote Notification Settings

Variable	Description
<code>/AS/EVENTSPRING/NOTIFICATION_SERVER1</code>	<p>primary notification server</p> <p>Format: host,agent_port,username,password where password is encrypted.</p> <p>Example: ns1,3181,patrolns,EA186757E5C6DA6</p>
<code>/AS/EVENTSPRING/NOTIFICATION_SERVER2</code>	<p>backup notification server</p> <p>Format: host,agent_port,username,password where password is encrypted.</p> <p>Example: ns2,3181,patrolns,EA186757E5C6DA6</p>
<code>/AS/EVENTSPRING/RemoteAgentCommSettings</code>	<p>agent-to-agent communication protocol</p> <p>Values are: TCP (default) UDP,timeout,retries</p> <p>Example: UDP,3</p>

Notification Server Settings

Table B-3 shows notification settings that you can apply to systems configured to perform notification. These settings only apply to notification servers.

Table B-3 Notification Server Settings

Variable	Description
/AS/EVENTSPRING/NS/nsRemoteTargetSetting	specifies how the notification server handles remote targets Values are: Merge (default) Override Ignore

Availability Monitor Settings

Table 2-4 shows the settings that you can apply to availability monitoring of hosts, nodes, PATROL Agents, and SNMP Agents.

Table 2-4 Availability Monitor Settings

Variable	Description
/AS/EVENTSPRING/AVAILABILITY/Primary	<p>primary availability monitor</p> <p>Once set, only the primary target is monitored. Backup availability monitors are used only when the primary is unavailable.</p> <p>Example: pismo 3181</p>
/AS/EVENTSPRING/AVAILABILITY/Targets /AS/EVENTSPRING/AVAILABILITY/Targets2	<p>agents and nodes to be monitored. Multiple targets are comma delimited.</p> <p>Format: hostname agent_port snmp_port</p> <p>The agent_port and snmp_port variables are optional and should be specified if you want PATROL Agent/SNMP Agent monitoring to be performed for the specified host.</p> <p>Example: boston 3181,ny - 161</p>
/AS/EVENTSPRING/AVAILABILITY/Blackouts/ <hostname>_<agent_port>_<snmp_port>	<p>comma delimited list of start and stop times for disabling availability checking on the target specified by (<hostname>_<agent_port>_<snmp_port>)</p> <p>The blackout period format is specified as <day> s1 s2 where s1 and s2 are the number of seconds past midnight.</p> <p>Example: Mon 3600 82800,Tue 3600 82800</p>

Table 2-4 Availability Monitor Settings

Variable	Description
/AS/EVENTSPRING/AVAILABILITY/checkerAccount	<p>user defined account used to perform PATROL Agent pings.</p> <p>Set on agents performing PATROL Availability Checking.</p> <p>The password is encrypted using PEM encryption.</p> <p>Example: patrolchecker,CDE867A57E5C6DA8</p>
/AS/EVENTSPRING/AVAILABILITY/pingCmd	<p>ping command to use when checking system availability</p> <p>Example: /usr/utls/myping.sh</p>
/AS/EVENTSPRING/AVAILABILITY/pingOkString	<p>the string that appears in the output of the pingCmd</p> <p>Example: bytes</p>
/AS/EVENTSPRING/AVAILABILITY/Updated	<p>a boolean flag that indicates whether targets or primary should be updated (refreshed)</p> <p>Values are: 0 = targets and primary are not automatically refreshed 1 = targets and primary are automatically refreshed</p>
/AS/EVENTSPRING/AVAILABILITY/Primary	<p>primary availability monitor</p> <p>Once set, only the primary target is monitored. Backup availability monitors are used only when the primary is unavailable.</p> <p>Example: pismo 3181</p>

Parameter Settings

Parameter settings only apply to parameter objects. The variable naming format is:

<CATEGORY>/<PARAMETER_OBJECT>/<VARIABLE>

For example:

/AS/EVENTSPRING/PARAM_SETTINGS/THRESHOLDS/CPU/CPU/CPUCpuUtil

Table 2-5 **Parameter Settings**

Variable	Description
Category: /AS/EVENTSPRING/PARAM_SETTINGS/POLLTIMES	
interval	polling time of a parameter (standard or collector parameters only) in seconds Example: 90
Category: /AS/EVENTSPRING/PARAM_SETTINGS/STATUSFLAGS	
paramSettingsStatusFlag	indicates whether PATROL KM for Event Management parameters should be used. Values are: 0 = use KM defaults 1 = use PATROL KM for Event Management parameter settings 2 = refresh thresholds 4 = refresh polling times 6 = refresh thresholds and polling times

Table 2-5 Parameter Settings

Variable	Description
Category: /AS/EVENTSPRING/PARAM_SETTINGS/THRESHOLDS	
	<p>parameter active status and alarm thresholds</p> <p>The THRESHOLDS setting comprises four comma-delimited values:</p> <ul style="list-style-type: none">• 1 – overall parameter active status• 2 – border thresholds• 3 – alarm range 1 thresholds• 4 – alarm range 2 thresholds <p>Threshold settings are determined as follows: active, min, max, when, N, and status where: active = 0 if inactive 1 if active min = lower range of threshold max = upper range of threshold when = when to alert 0 = immediately 1 = after N times 2 = after all Rrecovery actions failed N = number of times the threshold must be exceeded before an alert is triggered (used if 'when' = 1). status = desired alert status 0 = OK 1 = WARN 2 = ALARM</p> <p>Example: 1,1 0 100 0 0 2,1 1 50 0 0 1,1 51 100 0 0 2</p>

Application Class Settings

Application class settings apply only to application class objects. The variable naming format is:

<CATEGORY>/<CLASS_OBJECT>/<VARIABLE>

For example:

/AS/EVENTSPRING/APPLICATION_CLASSES/PRINTERS/active

Table B-6 Application Class Settings

Variable	Description
Category: /AS/EVENTSPRING/APPLICATION_CLASSES	
active	sets the active status of an application class
Note: Supported in 2.4.05.02 or later.	Values are: 0 = set active status to 0 (disable application class) 1 = set active status to 1 (perform pre-discovery for the application class) 2 = set active status to 2 (perform full discovery for the application class)
Category: /AS/EVENTSPRING/APPLICATION_CLASSESSTATUSFLAGS	
appClassSettingsStatusFlag	indicates whether PATROL KM for Event Management application class settings are used
Note: Supported in 2.4.05.02 or later.	Values are: 0 = use KM defaults 1 = use PATROL KM for Event Management application class settings 2 = refresh active rules

Menu Command Access Settings

You can use the Menu Command Access settings to control which PATROL KM for Event Management menu commands operators can execute from a PATROL Operator console.

Table B-7 Menu Command Access Settings

Variable	Description
Category: /AS/EVENTSPRING/MENU_COMMANDS	
allowOperator	Specifies which menu commands a user can execute from a PATROL Operator's Console.
Note: Supported in 2.5.00 or later.	Values are: <ul style="list-style-type: none">• all - all menu commands• availability - Availability menu commands• configNS - Quick Config -> Notification Server menu command• configRemoteAgent - Quick Config -> Remote Agent menu command• thresholds - Threshold menu commands• interval - Polltime menu commands• arsAction - Alert Action menu commands• arsCommand - Recovery Action menu commands• arsCmdType - Recovery Action Command Type menu commands• overrides - Override related menu commands• customId - Custom Identifier menu commands• alertLocalCommand - Notification Command menu command• alertSystem - Notification System menu command• alertResend - Alert Resend menu command• alertResetOnInit - SendResetOnInit menu command• msgText - Alert Messages menu commands• emailTargetsLocal - Local Email Target menu commands• emailTargetsRemote - Remote Email Target menu commands• pagerTargetsLocal - Local Pager Target menu commands• pagerTargetsRemote - Remote Pager Target menu commands• ttTargetsLocal - Local Trouble Ticket Target menu commands• ttTargetsRemote - Remote Trouble Ticket Target menu commands• customTargetsLocal - Local Custom Target menu commands• customTargetsRemote - Remote Custom Target menu commands• blackoutPeriod - Blackout Periods menu commands• instanceFiltering - Instance Filtering menu commands

All configuration changes take affect immediately once they have been applied. The following are exceptions:

- `arsAction`, only in certain circumstances
- `remoteCommSettings`

These configuration settings are updated by restarting the PATROL Agent after applying the configuration change. To affect changes without an agent restart, set the appropriate agent variables.

Sample Scenarios

Scenario: Send E-mail Notification for Alarm or Warning State

Your customers want e-mail notification when PATROL goes into a warning or alarm state. PATROL is currently running on two Solaris computers and three Windows NT computers. E-mails are sent from Unix. We will set up the Solaris computers as notification servers and the Windows NT computers as remote agents.

One of the Solaris computers is the primary notification server. This computer is responsible for sending an e-mail notification whenever an alert condition occurs on any of the computers. The other Solaris computer is the backup notification server. The backup notification server sends e-mail notifications whenever the primary notification server is unavailable. We will also create a default message format for notifications sent by the notification servers.

Assumptions

For this scenario we will assume the following:

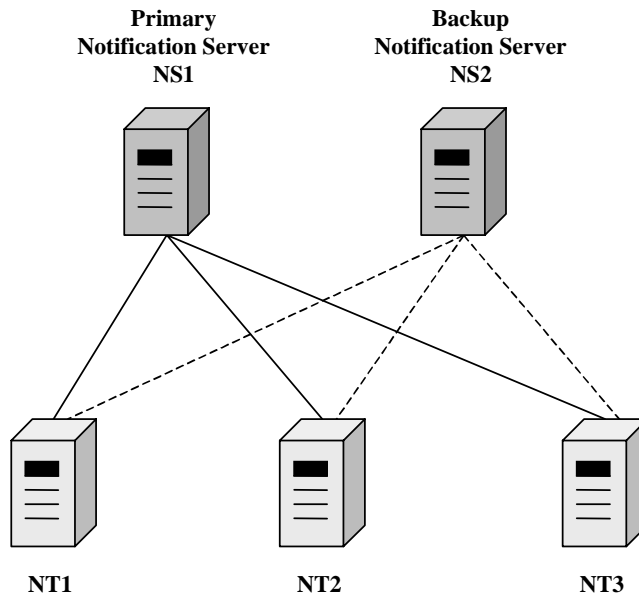
- A PATROL Developer Console for Windows is running on one of the Windows NT computers. The PATROL Developer Console is used for executing all PATROL KM for Event Management commands.

- There is a dedicated computer and agent for each notification server. This is the BMC Software recommended configuration.
- All PATROL Agents used for monitoring are running on port 3181.

Where to Start

Start by naming your computers. The two Solaris computers are defined as NS1 and NS2. These computers are the primary and the backup notification servers. Each computer must be loaded with PATROL KM for Event Management product. The NT computers are defined as NT1, NT2, and NT3. The diagram in Figure C-1 shows the configuration for this scenario.

Figure C-1 Scenario Send E-mail Notification for Alarm or Warning State Configuration



Step 1 Load the PATROL KM for Event Management

Load the KM from a PATROL Developer Console that has all of the PATROL Agents defined.

- 1.A** To load the KM, from the PATROL main menu, choose **File => Load KM.**

A dialog is displayed listing the available .kml files in the PATROL knowledge directory.

- 1.B** Select **EVENT_MANAGEMENT.kml**, then click **Open.**

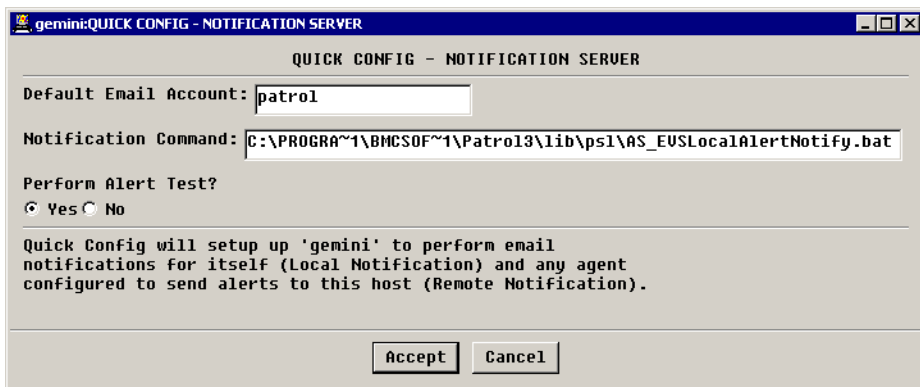
Step 2 Define the Notification Servers

You must configure both Notification Servers to perform notification.

- 2.A** Define the primary notification server, NS1, by right-clicking on NS1 on the PATROL Console, then choose **KM Commands => Event Management => Quick Config => Notification Server.**

The QUICK CONFIG - NOTIFICATION SERVER dialog (Figure C-2) is displayed.

Figure C-2 QUICK CONFIG - NOTIFICATION SERVER Dialog Box



- 2.B** Type the default e-mail account that receives notifications for all events that go into an alarm or warning state. When you define the actual notification target for each event that is triggered, the amount of e-mail going to the default e-mail account is reduced. Eventually, no mail is sent to this account. If any new e-mail is sent to this account, you must create a new rule to notify the appropriate person or group.
- 2.C** Type the file name of the script that is run for event notification. To avoid overwriting this script when an upgrade is performed, BMC Software recommends that the script be moved outside of the PATROL HOME directory. The default script provided with PATROL KM for Event Management is configured for mailx on Unix and Blat on Windows NT. If required, you must make any make any site modifications to the script.
- 2.D** If you want to perform an alert test to verify that a notification is sent to the default notification account, select **Yes**.
- 2.E** Click **Accept**. If an alert test was chosen, the test notification is sent. Verify that the e-mail was received by the default e-mail account.

Repeat Step 2 on page C-3 for the NS2 computer, which is the backup notification server.

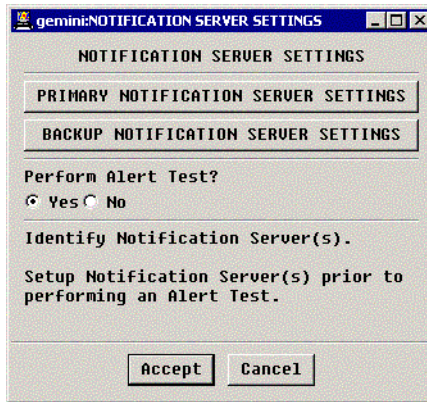
Step 3 Designate the primary and the backup notification servers

In this step, we designate NS1 as the primary notification server and NS2 as the backup notification server for the Windows NT computers (NT1, NT2, and NT3).

- 3.A** To designate NS1 as the primary notification server for NT1, right-click NT1 and choose **KM Commands => Event Management => Quick Config => Remote Agent**

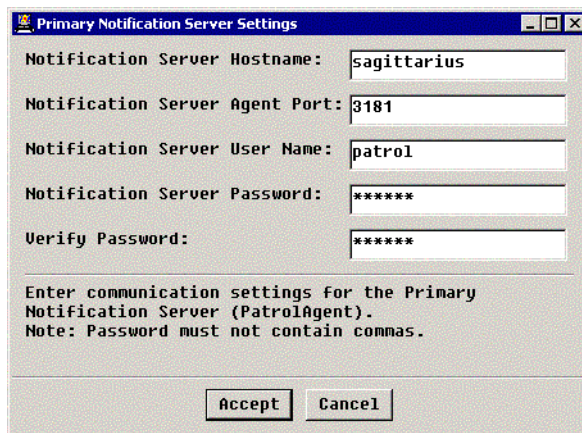
The NOTIFICATION SERVER SETTINGS dialog box (Figure C-2) displays.

Figure C-3 NOTIFICATION SERVER SETTINGS Dialog Box



- 3.B** Click **PRIMARY NOTIFICATION SERVER SETTINGS**. The Primary Notification Server Settings dialog box (Figure C-4 on page C-5) displays.

Figure C-4 Primary Notification Server Settings Dialog Box



- 3.C** Type **NS1**, the host name of the primary motification server, in the **Remote Hostname** field.

- 3.D** Type the userID you want to use for logging into the notification server in the **Remote User Name** field. The default for the **Remote User Name** field is the account you are currently logged in on.

Note

For better security, you must create a unique operating system account that is used only for remote notification on the notification servers. You can configure the notification login account to make it unable to fully login to the notification Sserver. For example, on Unix you can use an invalid login shell, such as **/bin/false** for the notification account.

- 3.E** Type the password for the notification login account in the **Remote Password** and **Verify Password** fields.
- 3.F** Click **Accept** to complete the configuration of the primary notification server and return to the REMOTE NOTIFICATION SERVER SETTINGS dialog box (Figure C-3 on page C-5).
- 3.G** Click **BACKUP NOTIFICATION SERVER SETTINGS**. The Backup Notification Server Settings dialog box (similar to Figure C-4 on page C-5) displays.
- 3.H** Type **NS2**, the host name of the backup notification server, in the **Remote Hostname** field.
- 3.I** Type the userID you want to use for logging into the notification server in the **Remote User Name** field. The default for the **Remote User Name** field is the account you are currently logged in on.

Note

For better security, you must create a unique operating system account that is used only for remote notification on the notification servers. You can configure the notification login account to make it unable to fully login to the notification server. For example, on Unix you can use an invalid login shell, such as **/bin/false** for the notification account.

- 3.J** Type the password for the notification login account in the **Remote Password** and **Verify Password** fields.

- 3.K** Click **Accept** to complete the configuration of the backup notification server and return to the REMOTE NOTIFICATION SERVER SETTINGS dialog box (Figure C-3 on page C-5).
- 3.L** If you want to perform a notification test for the NS1 computer, select **Yes** under **Perform Alert Test?**
- 3.M** Click **Accept** to complete the configuration of the primary and the backup notification servers and perform the notification test for the NS1 computer. Verify that the e-mail was received by the default e-mail account.

You can also verify the notification test by looking at the PATROL KM for Event Management NotifyEvents parameter. To verify:

1. Double-click the notification server, NS1. The AS_EVENTSPRING application class (Event Management) is displayed.
2. Double-click the AS_EVENTSPRING application class. The AS_EVENTSPRING application class parameters are displayed.
3. Double-click on the NotifyEvents parameter. If the test was successful, an annotated data point will be displayed for NS1.

Repeat Step 3 on page C-4 for the Windows NT computers: NT2 and NT3.

Step 4 Reword Messages

You can use PATROL KM for Event Management to create rules on the notification servers and the remote agents. Message Rewording rules reside on an individual agent, on a notification server, or on both. Creating and maintaining Message Rewording rules on the notification server centralizes the management of the rules. Creating these rules on the remote agents allows the groups responsible for the agents to create and maintain their own rules.

You can configure rules residing on both the notification server and the remote agents in several ways. By default, remote agent rules override the notification server rules. You can either set up a generic message format for all alerts on the notification server or you can create unique messages for specific applications, instances, or parameters.

In this scenario, you will create a generic message format that resides on the notification server. You will create this rule on NS1 and NS2.

- 4.A** Right-click the NS1 instance and choose **KM Commands => Event Management => Alert Setting => Alert Messages => Default Message Format**

The Message Rewording dialog box (Figure C-5) displays.

Figure C-5 Message Rewording Dialog Box

The screenshot shows a dialog box titled "geminiSET EVENT MANAGEMENT ALERT VARIABLES/RULES". Inside, there is a section labeled "SET EVENT MANAGEMENT ALERT VARIABLE:msgText". Below this, a prompt says "Enter Custom Message Text ('msgText'):" followed by a large text input field. A section titled "MESSAGE REWORDING OPTIONS" contains an example: "For example, if you entered: 'The Oracle server %APPINSTANCE% on %HOSTNAME% is consuming %PARAMETER_VALUE%% CPU (Ave CPU=%AVE10%%).'. The message you would see is: The Oracle server PROD1 on glamis is consuming 99.55% CPU (Ave CPU=76.54%).". At the bottom are "Accept" and "Cancel" buttons.

- 4.B** Type the message format you want to display using the message replacement variables. For example:

```
%PARAMETER_STATUS% on %HOSTNAME% for parameter %PARAMETER_NAME% =  
%PARAMETER_VALUE%
```

PATROL KM for Event Management substitutes real values at the script's run time. The following example shows a message that was received in an e-mail notification:

```
ALARM on NT1 for parameter CPUprcrProcesserTimePercent = 99.74
```

- 4.C** Click **Message Rewording Options** to view a list of replacement variables that can be used in the message. See Table 4-17, "Alert Messages Replacement Variables," on page 4-12 for a complete list and description of the message replacement variables provided by PATROL KM for Event Management.
- 4.D** Repeat Step 4 on page C-7 for the backup notification server, NS2.

Glossary

Note: In this glossary, the names of all Microsoft Windows products are referred to as *Microsoft Windows* or simply *Windows*. No distinctions are made between various Windows Servers, Windows 2000, and Windows NT.

access control list

A list that is set up by using a PATROL Agent configuration variable and that restricts PATROL Console access to a PATROL Agent. A PATROL Console can be assigned access rights to perform console, agent configuration, or event manager activities. The console server uses access control lists to restrict access to objects in the COS namespace.

agent namespace

See PATROL Agent namespace.

Agent Query

A PATROL Console feature that constructs SQL-like statements for querying PATROL Agents connected to the console. Agent Query produces a tabular report that contains information about requested objects and can be used to perform object management activities, such as disconnecting and reconnecting computers. Queries can be saved, reissued, added, or changed. PATROL offers built-in queries in the Quick Query command on the Tools menu from the PATROL Console main menu bar. *See also* Quick Query.

alarm	An indication that a parameter for an object has returned a value within the alarm range or that application discovery has discovered that a file or process is missing since the last application check. An alarm state for an object can be indicated by a flashing icon, depending on the configuration of a console preference. <i>See also</i> warning.
alert	A PATROL object changing status. For example, Ok to Alarm. An alert condition is typically caused by a parameter exceeding a threshold.
alert range	A range of values that serve as thresholds for a warning state or an alarm state. Alert range values cannot fall outside of set border range values. <i>See also</i> border action, border range, and recovery action.
ALL_COMPUTERS class	The highest-level computer class in PATROL. Attributes assigned to this class will be inherited by all computer classes known to PATROL. <i>See also</i> class and computer class.
annotated data point	A specially marked point on a parameter graph that provides detailed information about a parameter at a particular moment. The associated data is accessed by double-clicking the data point, which is represented by a user-specified character (the default is an asterisk) in PATROL 3.x and earlier or by a set bitmap in PATROL Central 7.x. <i>See also</i> parameter.
application account	An account that you define at KM setup and that you can change for an application class or instance. An application account is commonly used to connect to an RDBMS on a server where the database resides or to run SQL commands.
application check cycle	The interval at which application discovery occurs. The PATROL Agent process cache (as opposed to the system process table) is checked to ensure that all application instances and files previously discovered still exist there. <i>See also</i> application discovery, application discovery rules, prediscovery, process cache refresh, PSL discovery, and simple discovery.

application class	The object class to which an application instance belongs; also, the representation of the class as a container (Unix) or folder (Windows) on the PATROL Console. You can use the developer functionality of a PATROL Console to add or change application classes. <i>See also</i> class.
application discovery	A PATROL Agent procedure carried out at preset intervals on each monitored computer to discover application instances. When an instance is discovered, an icon appears on the PATROL interface. The application class includes rules for discovering processes and files by using simple process and file matching or PSL commands. Application definition information is checked against the information in the PATROL Agent process cache, which is periodically updated. Each time the PATROL Agent process cache is refreshed, application discovery is triggered. <i>See also</i> application check cycle, application discovery rules, PATROL Agent process cache, prediscovery, PSL discovery, and simple discovery.
application discovery rules	A set of rules stored by the PATROL Agent and periodically evaluated to find out whether a specific instance of an application class exists in the monitored environment. The rules describe how a PATROL Agent can detect instances of the application on a computer. There are two types of application discovery: simple and PSL; PSL discovery can include prediscovery rules as well as discovery rules. <i>See also</i> application check cycle, application discovery, prediscovery, PSL discovery, and simple discovery.
application filter	A feature used from the PATROL Console to hide all instances of selected application classes for a particular computer. The PATROL Agent continues to monitor the application instances by running parameter commands and recovery actions.
application instance	A system resource that is discovered by PATROL and that contains the information and attributes of the application class that it belongs to. <i>See also</i> application class and instance.

application state	<p>The condition of an application class or an application instance. The most common application states are OK, warning, and alarm. An application class or instance icon can also show additional conditions. <i>See also</i> computer state and parameter state.</p>
attribute	<p>A characteristic that is assigned to a PATROL object (computer class, computer instance, application class, application instance, or parameter) and that you can use to monitor and manage that object. Computers and applications can have attributes such as command type, parameter, menu command, InfoBox command, PATROL setup command, state change action, or environment variable. Parameters can have attributes such as scheduling, command type, and thresholds.</p> <p>An attribute can be defined globally for all instances of a class or locally for a particular computer or application instance. An instance inherits attributes from a class; however, an attribute defined at the instance level overrides inherited attributes. <i>See also</i> global level and local level.</p>
border action	<p>A command or recovery action associated with a parameter border range and initiated when that range has been breached. Border actions can be initiated immediately when the parameter returns a value outside the border range, after a warning or alarm has occurred a specified number of times, or after all other recovery actions have failed. <i>See also</i> border range.</p>
border range	<p>A range of values that serve as thresholds for a third-level alert condition when it is possible for a parameter to return a value outside of the alarm range limits. When a border range is breached, border actions can be initiated. <i>See also</i> border action.</p>
built-in command	<p>An internal command available from the PATROL Agent that monitors and manages functions such as resetting the state of an object, refreshing parameters, and echoing text. The command is identified by the naming convention %command_name. <i>See also</i> built-in macro variable.</p>

built-in macro variable	An internal variable created and maintained by PATROL for use in built-in commands and PSL. The naming convention for the variable is <code>%{variable_name}</code> . <i>See also</i> built-in command.
chart	A plot of parameter data values made by the PATROL Console Charting Server. <i>See also</i> multigraph container and PATROL Console Charting Server.
charting server	<i>See</i> PATROL Console Charting Server.
class	The object classification in PATROL where global attributes can be defined; the attributes are then inherited by instances of the class. An instance belongs to a computer class or an application class. <i>See also</i> application class, computer class, and event class.
collector parameter	A type of parameter that contains instructions for gathering values for consumer parameters to display. A collector parameter does not display any value, issue alarms, or launch recovery actions. <i>See also</i> consumer parameter, parameter, and standard parameter.
command line argument	An option for starting a PATROL Agent or a PATROL Console at the operating system command line. PATROL Agent arguments include names of KMs to load and port numbers for agent-console connection. PATROL Console arguments include connection mode (developer or operator), user ID to start the PATROL Console, names of KMs to load, and names of the files to use.
command line interface	<i>See</i> PATROL Command Line Interface (CLI).
command text editor	The component that provides basic text editing functions for a PATROL Console. It is commonly used to add or change commands (menu commands, parameter data collection and recovery actions, InfoBox commands, setup commands, and state change actions).

command type	The designation assigned to a command according to its manner of execution. This attribute must be defined for a parameter command, a parameter recovery action, a menu command, an InfoBox command, a setup command, or a state change action. The PATROL Agent provides two command types: operating system (OS) and PSL. PATROL KMs provide additional command types. The developer functionality of a PATROL Console can be used to add or change command types.
commit	The process of saving to PATROL Agent computers the changes that have been made to a KM by using a PATROL Console. A PATROL user can disable a PATROL Console's ability to commit KM changes.
computer class	The basic object class to which computer instances of the same type belong. Examples include Solaris, OSF1, HP, and RS6000. PATROL provides computer classes for all supported computers and operating systems; a PATROL Console with developer functionality can add or change computer classes.
computer instance	A computer that is running in an environment managed by PATROL and that is represented by an icon on the PATROL interface. A computer instance contains the information and attributes of the computer class that it belongs to. <i>See also</i> instance.
computer state	The condition of a computer. The main computer states are OK, warning, and alarm. A computer icon can show additional conditions that include no output messages pending, output messages pending, void because a connection cannot be established, and void because a connection was previously established but now is broken. <i>See also</i> state.
configuration file, KM	<i>See</i> KM configuration file.
configuration file, PATROL Agent	<i>See</i> PATROL Agent configuration file.

connection mode	The mode in which the PATROL Console is connected to the PATROL Agent. The mode can be developer or operator and is a property of the Add Host dialog box (PATROL 3.x and earlier), an Add Managed System wizard, or other connecting method. The connection mode is a global (console-wide) property that can be overridden for a computer instance. <i>See also</i> PATROL Console.
console module	A program that extends the functionality of PATROL Central or PATROL Web Central. Console modules can collect data, subscribe to events, access Knowledge Module functions, authenticate users, and perform security-related functions.
console server	A server through which PATROL Central and PATROL Web Central communicate with managed systems. A console server handles requests, events, data, communications, views, customizations, and security.
consumer parameter	A type of parameter that displays a value that was gathered by a collector parameter. A consumer parameter never issues commands and is not scheduled for execution; however, it has alarm definitions and can run recovery actions. <i>See also</i> collector parameter, parameter, and standard parameter.
container	A custom object that you can create to hold any other objects that you select—such as computers, applications, and parameters—in a distributed environment. In Windows, a container is referred to as a folder. You can drag and drop an object into and out of a container icon. However, objects from one computer cannot be dropped inside another computer. Once a container is defined, the object hierarchy applies at each level of the container. That is, a container icon found within a container icon assumes the variable settings of the container in which it is displayed. <i>See also</i> object hierarchy and PATROL Console Charting Server.
customize a KM	To modify properties or attributes locally or globally. <i>See also</i> global level and local level.
customize a parameter	<i>See</i> override a parameter.

custom view	A grid-like view that can be created in PATROL Central or PATROL Web Central to show user-selected information.
deactivate a parameter	To stop running a parameter for selected computer or application instances. In PATROL Consoles for Microsoft Windows environments, deactivating a parameter stops parameter commands and recovery actions and deletes the parameter icon from the application instance window without deleting the parameter definition in the KM tree. A deactivated parameter can be reactivated at any time. <i>See also</i> snooze an alarm and suspend a parameter.
deactivate an application class	To stop monitoring an application class and all of its instances on selected computer instances. In PATROL Consoles for Microsoft Windows environments, deactivating an application class deletes the application class and all of its instance icons from the computer window without deleting the application class or definition in the KM tree. A deactivated application class can be reactivated at any time. <i>See also</i> application filter and deactivate a parameter.
desktop file	In PATROL 3.x and earlier, a file that stores your desktop layout, the computers that you monitor, the KMs that you loaded, and your PATROL Console user accounts for monitored objects. You can create multiple desktop files for any number of PATROL Consoles. By default, desktop files always have a .dt extension. Desktop files are replaced by management profiles in PATROL 7.x. <i>See also</i> desktop template file.
desktop template file	In PATROL 3.x and earlier, a file that stores information about the desktop setup of one computer. You can create multiple desktop template files for any number of PATROL Consoles. Each PATROL Console user can apply a template to selected computers on the desktop. By default, desktop template files always have a .dtm extension. <i>See also</i> desktop file.
Desktop tree	<i>A feature of PATROL for Microsoft Windows only.</i> One of the views of folders available with PATROL for Microsoft Windows environments, the Desktop tree displays the object hierarchy. <i>See also</i> KM tree.

developer mode

An operational mode of the PATROL Console that can be used to monitor and manage computer instances and application instances and to customize, create, and delete locally loaded Knowledge Modules and commit these changes to selected PATROL Agent computers. *See also* PATROL Console.

**disable an application,
disable a KM**

To temporarily or permanently block an application or KM from loading and to block the PATROL Agent from using that KM. When a KM is disabled (added to the disabled list) in the agent configuration file, the KM files are not deleted from the PATROL Agent computers, but the PATROL Agent stops using the KM to collect parameter data and run recovery actions. The default is that no KMs are disabled. Most KMs are composed of individual application files with a **.km** extension. *See also* preloaded KM, static KM, and unload a KM.

discovery

See application discovery.

**distribution CD or
tape**

A CD or tape that contains a copy of one or more BMC Software products and includes software and documentation (user guides and online help systems).

environment variable

A variable used to specify settings, such as the program search path for the environment in which PATROL runs. You can set environment variables for computer classes, computer instances, application classes, application instances, and parameters.

event

The occurrence of a change, such as the appearance of a task icon, the launch of a recovery action, the connection of a console to an agent, or a state change in a monitored object (computer class, computer instance, application class, application instance, or parameter). Events are captured by the PATROL Agent, stored in an event repository file, and forwarded to an event manager (PEM) if an event manager is connected. The types of events forwarded by the agent are governed by a persistent filter for each event manager connected to a PATROL Agent.

**event
acknowledgment
command**

A command that is triggered by the PATROL Agent when an event is acknowledged in an event manager (PEM). *See also* event escalation command and event notification command.

event catalog	A collection of event classes associated with a particular application. PATROL provides a Standard Event Catalog that contains predefined Standard Event Classes for all computer classes and application classes. You can add, customize, and delete an application event catalog only from a PATROL Console in the developer mode. <i>See also</i> event class and Standard Event Catalog.
event class	A category of events that you can create according to how you want the events to be handled by an event manager and what actions you want to be taken when the event occurs. Event classes are stored in event catalogs and can be added, modified, or deleted only from a PATROL Console in the developer mode. PATROL provides a number of event classes in the Standard Event Catalog, such as worst application and registered application. <i>See also</i> event catalog and Standard Event Catalog.
event class command	A command that is run by the PATROL Agent when certain events occur and that is used in conjunction with an event manager (PEM). The commands are specified for the event class that the event is associated with. A command can be one of three types: escalation, notification, or acknowledgment. <i>See also</i> event acknowledgment command, event escalation command, and event notification command.
Event Diary	The part of an event manager (PEM) where you can store or change comments about any event in the event log. You can enter commands at any time from the PATROL Event Manager Details window.
event escalation command	A command that is triggered by the PATROL Agent when an event is not acknowledged, closed, or deleted within an event manager (PEM) by the end of the escalation period. <i>See also</i> event acknowledgment command, event escalation period, and event notification command.

event escalation period	A period during which the severity of an event is increased as a result of the event's persistence. Escalation actions are part of escalation command definitions for event classes and can be triggered only by the PATROL Agent. <i>See also</i> event escalation command.
event history repository	A circular file where events are stored by the PATROL Agent and accessed by an event manager, such as the PEM. The file resides on the PATROL Agent computer and retains a limited number of events. When the maximum number of events is reached and a new event is stored, the oldest event is removed in a cyclical fashion. <i>See also</i> parameter history repository.
event manager	A graphical user interface for monitoring and managing events. The event manager can be used with or without the PATROL Console. <i>See also</i> PATROL Event Manager (PEM).
event notification command	A command that is triggered by the PATROL Agent when an event is logged into an event manager (PEM). <i>See also</i> event acknowledgment command and event escalation command.
event type	The PATROL-provided category for an event according to a filtering mechanism in an event manager. Event types include information, state change, error, warning, alarm, and response.
event view filter	<i>See</i> view filter.
event-driven scheduling	A kind of scheduling that starts a parameter when certain conditions are met. <i>See also</i> periodic scheduling.
expert advice	Comments about or instructions for dealing with PATROL events as reported by the agent. Expert advice is defined in the Event Properties dialog box in a PATROL Console in the developer mode.
filter, application	<i>See</i> application filter.
filter, event view	<i>See</i> view filter.
filter, persistent	<i>See</i> persistent filter.

global channel	A single dedicated connection through which PATROL monitors and manages a specific program or operating system. The PATROL Agent maintains this connection to minimize the consumption of program or operating system resources.
global level	In PATROL hierarchy, the level at which object properties and attributes are defined for all instances of an object or class. An object at the local level inherits characteristics (properties) and attributes from the global level. <i>See also</i> local level.
heartbeat	A periodic message sent between communicating objects to inform each object that the other is still “alive.” For example, the PATROL Console checks to see whether the PATROL Agent is still running.
heartbeat interval	The interval (in seconds) at which heartbeat messages are sent. The longer the interval, the lower the network traffic. <i>See also</i> message retries, message time-out, and reconnect polling.
history	Parameter and event values that are collected and stored on each monitored computer. Parameter values are stored in binary files for a specified period of time; events are stored in circular log files until the maximum size is reached. The size and location of parameter history files are specified through either the PATROL Console or the PATROL Agent; size and location of event history files are specified through an event manager, such as the PEM, or the PATROL Agent.
history repository	A binary file in which parameter values (except those that are displayed as text) are stored by the PATROL Agent and accessed by the PATROL Console for a specified number of days (the default is one day). When the number of storage days is reached, those values are removed in a cyclical fashion.
history retention level	The specified level (global or local) where the parameter history retention period for an object is set. The period can be inherited from the next higher level in the object hierarchy or set at the local level. If the history retention level is local, the number of days that history is stored (retention period) must be set. <i>See also</i> history retention period.

history retention period

The number of days that parameter values are stored in the history database before they are automatically purged by PATROL. The period can be specified at the class (global) or instance (local) level. History retention can be set for all parameters of a computer class, a computer instance, an application class, or an application instance. History for an individual parameter on an application instance can be manually cleared at any time by using a PATROL Console. *See also* history retention level.

history span

The combined settings for a parameter's history retention level and history retention period. *See also* history retention level and history retention period.

InfoBox

A dialog box that contains a static list of fields and displays current information about an object, such as the version number of an RDBMS and whether the object is online or offline. Commands are run when the InfoBox is opened. Information can be manually updated if the InfoBox remains open for a period of time. PATROL provides a number of commands for obtaining and displaying object information in an InfoBox. Only a PATROL Console in the developer mode can be used to add or change commands.

information event

Any event that is not a state change or an error. Typical information events occur when a parameter is activated or deactivated, a parameter is suspended or resumed, or application discovery is run. The default setting for PATROL is to prevent this type of event from being stored in the event repository. To store and display this type of event, you must modify the persistent filter setting in the PATROL Agent configuration file.

instance

A computer or discovered application that is running in an environment managed by PATROL. An instance has all the attributes of the class that it belongs to. A computer instance is a monitored computer that has been added to the PATROL Console. An application instance is discovered by PATROL. *See also* application discovery, application instance, and computer instance.

KM	<i>See</i> Knowledge Module (KM).
KM configuration file	A file in which the characteristics of a KM are defined through KM menu commands during KM installation and setup (if setup is required). <i>See also</i> Knowledge Module (KM) and PATROL Agent configuration file.
KM list	A list of KMs used by a PATROL Agent or PATROL Console. <i>See also</i> Knowledge Module (KM).
KM Migrator	<i>See</i> PATROL KM Migrator. <i>See also</i> Knowledge Module (KM).
KM package	<i>See</i> Knowledge Module package.
KM tree	<i>A feature of PATROL for Microsoft Windows only.</i> One of two views of folders available in Windows. The KM tree displays computer classes, application classes, and their customized instances in the knowledge hierarchy and also displays the Standard Event Catalog. A PATROL Console in operator mode can only view the KM tree; only a PATROL Console in the developer mode can change KM properties and attributes. <i>See also</i> Desktop tree and Knowledge Module (KM).
knowledge hierarchy	The rules by which objects inherit or are assigned attributes. (In PATROL Consoles for Microsoft Windows environments, classes of objects are represented in the Computer Classes and Application Classes sets of folders on the KM tree.) Properties and attributes of a customized instance override those defined for the class to which the instance belongs.

Knowledge Module (KM)

A set of files from which a PATROL Agent receives information about resources running on a monitored computer. A KM file can contain the actual instructions for monitoring objects or simply a list of KMs to load. KMs are loaded by a PATROL Agent and a PATROL Console.

KMs provide information for the way monitored computers are represented in the PATROL interface, for the discovery of application instances and the way they are represented, for parameters that are run under those applications, and for the options available on object pop-up menus. A PATROL Console in the developer mode can change KM knowledge for its current session, save knowledge for all of its future sessions, and commit KM changes to specified PATROL Agent computers. *See also* commit, KM configuration file, KM list, KM Migrator, KM tree, load KMs, and version arbitration.

Knowledge Module package

A package of PATROL KM files that can be distributed by an installation program or stored in and distributed by the PATROL KMDS. The package file has a **.pkg** file extension. KM packages are created by using a PATROL Console in the developer mode. *See also* Knowledge Module (KM), PATROL Console, PATROL Knowledge Module Deployment Server (PATROL KMDS), and PATROL Knowledge Module Deployment Server Manager (PATROL KMDS Manager).

load applications

Same as load KMs. Most KMs are composed of application files with a **.km** extension.

load KMs

To place KM files into memory for execution. After configuration and during startup, the PATROL Agent loads the KM files that are listed in its configuration file and that reside on the PATROL Agent computer. When a PATROL Console connects to the PATROL Agent, the KM versions that the agent executes depend on whether the console has developer or operator functionality. *See also* Knowledge Module (KM) and version arbitration.

local history

The history (stored parameter values) for an object or instance. *See also* global level and local level.

local history retention period	The length of time set by the user during which stored parameter values for an object or instance are retained.
local level	In PATROL hierarchy, the level of a computer instance or an application instance. An object (instance) at the local level inherits properties and attributes that are defined globally. When properties and attributes are customized locally for an individual instance, they override inherited attributes. <i>See also</i> global level.
managed object	Any object that PATROL manages. <i>See</i> object.
managed system	A system—usually a computer on which a PATROL Agent is running—that is added (connected) to a PATROL Console to be monitored and managed by PATROL and that is represented by an icon on the PATROL interface.
management profile	A user profile for PATROL Central and PATROL Web Central that is stored by the console server. A management profile is similar to a session file and contains information about custom views, your current view of the PATROL environment, information about systems that you are currently managing, Knowledge Module information, and console layout information for PATROL Central. Management profiles replace desktop files and session files that were used in PATROL 3.x and earlier.
master agent	<i>See</i> PATROL SNMP Master Agent.
message retries	<i>A feature of UDP only.</i> The number of times that the PATROL Console will re-send a message to the PATROL Agent. The greater the number of message retries, the more time the PATROL Console will give the PATROL Agent to respond before deciding that the agent connection is down and timing out. The number of message retries multiplied by message time-out (in seconds) is the approximate time allowed for a connection verification. <i>See also</i> heartbeat, heartbeat interval, message time-out, and reconnect polling.

message time-out	<i>A feature of UDP only.</i> The time interval (in seconds) that the PATROL Console will give the PATROL Agent to respond to a connection verification before deciding that the Agent connection is down. The number of message retries multiplied by message time-out is the approximate time allowed for a connection verification. <i>See also</i> heartbeat, heartbeat interval, message retries, and reconnect polling.
message window	A window that displays command output and error messages from the PATROL Console graphical user interface. <i>See also</i> response window, system output window, and task output window.
multigraph container	A custom object into which you can drop parameter objects to be plotted as charts. <i>See also</i> PATROL Console Charting Server.
notification	An action in response to a PATROL event. Notifications can include pages, e-mails, trouble tickets, and pop-up windows.
object	A computer class, computer instance, application class, application instance, parameter, or container (folder) in an environment managed by PATROL. Objects have properties and are assigned attributes (command types, parameters, menu commands, InfoBox commands, setup commands, state change actions, and environment variables). Parameter objects use data collection commands to obtain values from classes and instances. <i>See also</i> object class, object hierarchy, object icon, and object window.
object class	A computer class or application class. <i>See also</i> class, object, and object hierarchy.
object hierarchy	The structure of object levels in PATROL. On the PATROL interface, computers contain application folders (containers) representing a loaded KM, application folders contain one or more application instances, and application instances contain parameters.

object icon	A graphic that represents a computer instance, application class, application instance, parameter, or container (folder) in an environment managed by PATROL. <i>See also</i> object, object hierarchy, and object window.
object window	An open object container (folder) that may contain application class icons, application instance icons, parameter icons, custom containers (folders), and shortcuts. The object window is displayed when you double-click the object icon. <i>See also</i> application instance, computer instance, object, and object icon.
operator mode	An operational mode of the PATROL Console that can be used to monitor and manage computer instances and application instances but not to customize or create KMs, commands, and parameters. <i>See also</i> PATROL Console.
operating system account	An account that is set up at installation to grant the PATROL Agent access to a computer. Operating system commands executed by the PATROL Agent and PATROL Console use this account. The PATROL Agent configuration specifies a default operating system account, which can be changed.
override a parameter	To disable or change the behavior of a local PATROL application parameter. The changes to the parameter are local to the managed system running the parameter and are stored in the agent configuration database. You must be granted specific permissions by a PATROL Administrator through the PATROL User Roles file in order to override parameters. Override a parameter is replaced by customize a parameter in PATROL 7.x. <i>See also</i> PATROL roles.

parameter	The monitoring element of PATROL. Parameters are run by the PATROL Agent; they periodically use data collection commands to obtain data on a system resource and then parse, process, and store that data on the computer that is running the PATROL Agent. Parameters can display data in various formats, such as numeric, text, stoplight, and Boolean. Parameter data can be accessed from a PATROL Console, PATROLVIEW, or an SNMP console. Parameters have thresholds and can trigger warnings and alarms. If the value returned by the parameter triggers a warning or an alarm, the PATROL Agent notifies the PATROL Console and runs any recovery actions associated with the parameter. <i>See also</i> parameter history repository and parameter state.
parameter cache	The memory location where current parameter data is kept. In the PATROL Agent's configuration file, you can set the size of the cache, the maximum number of data points that can be stored, and the interval (in seconds) for emptying the cache.
parameter history repository	Also known as parameter history file. <i>See</i> history repository.
parameter override	<i>See</i> override a parameter.
parameter state	The condition of a parameter. The most common parameter states are OK, warning, and alarm. In PATROL 3.x and earlier, a parameter icon can show additional conditions that include no history, offline, and suspended. In PATROL 7.x, the suspended state is shown in the label—for example, MyParam (suspended)—rather than in the icon. A parameter can also be deactivated; when a parameter is deactivated, no icon is displayed. <i>See also</i> state.
PATROL Agent	The core component of PATROL architecture. The agent is used to monitor and manage host computers and can communicate with the PATROL Console, a stand-alone event manager (PEM), PATROLVIEW, and SNMP consoles. From the command line, the PATROL Agent is configured by the pconfig utility; from a graphical user interface, it is configured by the xpconfig utility for Unix or the wpconfig utility for Windows. <i>See also</i> PATROL SNMP Master Agent.

PATROL Agent configuration file	A file in which you can define the characteristics of the PATROL Agent by setting PATROL Agent configuration variables. You can edit the configuration file by using the pconfig utility, the wpconfig utility, or the xpconfig utility. <i>See also</i> KM configuration file, PATROL Agent configuration variable, pconfig, wpconfig, and xpconfig.
PATROL Agent configuration variable	The means by which the characteristics of a PATROL Agent are defined. PATROL provides default variable values that can be customized. Configuration variables determine such characteristics as how errors are handled, which KMs are loaded and how, how SNMP support is configured, and how events trigger SNMP traps. <i>See also</i> PATROL Agent configuration file.
PATROL Agent Manager	<i>A feature of PATROL for Microsoft Windows only.</i> The graphical user interface used to install and run the PATROL Agent.
PATROL Agent namespace	A memory array that contains an internal representation of the PATROL object hierarchy. Values in the agent namespace are available to PSL scripts, eliminating the need to develop code to collect this data.
PATROL Agent process cache	A snapshot of the operating system process table on a monitored computer. The agent process cache is updated periodically.
PATROL Agent process cache refresh	A periodic process of the PATROL Agent that issues a platform-dependent system query to obtain a list of the active processes. This data is used to update the PATROL Agent process cache.
PATROL Agent run queue	A time-ordered schedule of actions, such as application discovery and parameter execution, to be carried out by the PATROL Agent. <i>See also</i> PSL run queue.

PATROL Command Line Interface (CLI)

An interface program that you can access from the command line of a monitored computer and through which you can run some PATROL products and utilities. With the CLI, you can monitor the state of PATROL Agents remotely, execute PSL functions, and query and control events. The CLI is used in place of the PATROL Console when memory and performance constraints exist.

PATROL Console

The graphical user interface from which you launch commands and manage the environment monitored by PATROL. The PATROL Console displays all of the monitored computer instances and application instances as icons. It also interacts with the PATROL Agent and runs commands and tasks on each monitored computer. The dialog is event-driven so that messages reach the PATROL Console only when a specific event causes a state change on the monitored computer.

A PATROL Console with developer functionality can monitor and manage computer instances, application instances, and parameters; customize, create, and delete locally loaded Knowledge Modules and commit these changes to selected PATROL Agent computers; add, modify, or delete event classes and commands in the Standard Event Catalog; and define expert advice. A PATROL Console with operator functionality can monitor and manage computer instances, application instances, and parameters and can view expert advice but not customize or create KMs, commands, and parameters. *See also* developer mode and operator mode.

PATROL Console Charting Server

A PATROL function that creates charts and graphs of actual values returned by more than one parameter. Charts and graphs are created by dragging and dropping various parameters into a multigraph container (folder) and plotting the results into a chart. Parameter data is plotted either in real time or from history sets and can be presented in a number of chart styles, including line graphs, pie charts, 3-D bar charts, and area plots. Charts can be viewed through the PATROL Console and printed to a local printer or PostScript file.

PATROL Enterprise Manager (PATROL EM)

An event management system that gathers, filters, translates, and prioritizes messages from the managed systems in an enterprise and displays them as alerts in a single console. The PATROL EM consolidates alerts from different vendors and different geographical locations into a single display for fast identification and resolution of potential problems.

PATROL Event Manager (PEM)

An event manager that you can use to view and manage events that occur on monitored system resources and that are sent by PATROL Agents. You can access the PEM from the PATROL Console or use it as a stand-alone facility. It works with the PATROL Agent and user-specified filters to provide a customized view of events. *See also* event manager.

PATROL History Loader KM

A PATROL utility used to convert PATROL parameter history data into an ASCII data file or to store parameter history data directly into a particular relational database management system.

PATROLVIEW

Formerly PATROLVIEW or PATROLINK. Products that can be used to view events and to monitor and display all the parameters provided by the PATROL Agents and KMs in a network or enterprise management console.

PATROL KMDS

See PATROL Knowledge Module Deployment Server (PATROL KMDS).

PATROL KMDS Manager

See PATROL Knowledge Module Deployment Server Manager (PATROL KMDS Manager).

PATROL KM Migrator

A PATROL utility used to propagate KM user customizations to newly released versions of PATROL Knowledge Modules.

PATROL Knowledge Module Deployment Server (PATROL KMDS)

The change and version control tool for KMs. A repository for storage of PATROL KMs and changes to those KMs.

**PATROL Knowledge
Module Deployment
Server Manager
(PATROL KMDS
Manager)**

The graphical interface for the PATROL KMDS that can be used to manage and deploy or distribute KM changes in the production environment.

PATROL roles

In PATROL 3.x and earlier, a set of permissions that grant or remove the ability of a PATROL Console or PATROL Agent to perform certain functions. PATROL roles are defined in the PATROL User Roles file, which is read when the console starts.

**PATROL Script
Language (PSL)**

A scripting language (similar to Java) that is used for generic system management and that is compiled and executed on a virtual machine running inside the PATROL Agent. PSL is used for writing application discovery procedures, parameters, recovery actions, commands, and tasks for monitored computers within the PATROL environment.

**PATROL SNMP Master
Agent**

The agent through which a PATROL Agent interacts with an SNMP agent and SNMP manager. The PATROL Master Agent configuration file contains the community name and port number for all agents in such a multiple-agent architecture.

patroldev

In PATROL 3.x and earlier, a domain group that can be set up by a Windows system administrator to restrict user access to a PATROL Developer Console. When a user tries to start a PATROL Console with developer functionality, PATROL checks whether the user is in the patroldev group. If the user is not in the group, a PATROL Console with operator functionality is started instead. *See also* ptrldev.

pconfig

The command line utility for setting PATROL Agent configuration variables. *See also* PATROL Agent configuration file, PATROL Agent configuration variable, wpconfig, and xpconfig.

PEM

See PATROL Event Manager (PEM).

periodic scheduling

A kind of scheduling that starts a parameter at a certain time and reruns the parameter at certain intervals. *See also* event-driven scheduling.

persistent filter	A filter maintained by the PATROL Agent for each PATROL Console or event manager that connects to it. The filter is used to minimize network traffic by limiting the number and types of events that are forwarded from a PATROL Agent to a PATROL Console or an event manager (PEM).
polling cycle	The schedule on which a parameter starts running and the intervals at which it reruns; the cycle is expressed in seconds. <i>See also</i> event-driven scheduling and periodic scheduling.
pop-up menu	The menu of commands for a monitored object; the menu is accessed by right-clicking the object.
prediscovery	A quick one-time test written in PSL to determine whether a resource that you want to monitor is installed or running on a monitored computer. If the results are affirmative, the PATROL Agent runs the discovery script. Prediscovery helps reduce PATROL Agent processing requirements.
preloaded KM	A KM that is loaded by the PATROL Agent at startup and run as long as the Agent runs. <i>See also</i> disable an application, disable a KM and static KM.
process cache refresh	<i>See</i> PATROL Agent process cache refresh.
property	A characteristic or attribute of an object, such as its icon.
PSL	<i>See</i> PATROL Script Language (PSL).
PSL Compiler	A PATROL utility that compiles PSL scripts into a binary byte code that can be executed by the PSL virtual machine. The PSL Compiler can also be used to check a PSL script for syntax errors. The compiler is embedded in the PATROL Agent and PATROL Console (PATROL 3.x and earlier) and can also be run as a command-line utility.
PSL Debugger	A PATROL Console utility that is used to debug PSL scripts. The PSL debugger is accessed through a computer's pop-up menu.

PSL discovery	A type of application discovery in which the discovery rules are defined by using PSL. PSL discovery can consist of prediscovery and discovery PSL scripts.
PSL Profiler	A PATROL utility that is used to tune the CPU usage and minimize child processes or file operations of a newly created KM. When the PSL Profiler is enabled, the PATROL Agent starts accumulating and recording profile statistics.
PSL run queue	A queue of the currently executing PSL processes. The PSL run queue is used to distribute processing time between PSL processes in a round-robin fashion.
ptrlddev	In PATROL 3.x and earlier, a form of patroldev that can be used in environments that support domain names no larger than eight characters. <i>See also</i> patroldev.
Quick Query	In PATROL 3.x and earlier, a command on the Tools menu from the PATROL Console main menu bar that contains built-in predefined commands that you can use to query the agent for frequently needed information. For example, you can query the agent regularly about all computer instances, application instances, and parameters that are in a warning or alarm state. <i>See also</i> Agent Query.
reconnect polling	The time interval (in seconds) at which the PATROL Console will try to reconnect to a PATROL Agent that has dropped the previous connection. The longer the interval, the lower the network traffic. <i>See also</i> heartbeat, heartbeat interval, message retries, message time-out.
recovery action	A procedure that attempts to fix a problem that caused a warning or alarm condition. A recovery action is defined within a parameter by a user or by PATROL and triggered when the returned parameter value falls within a defined alarm range.

refresh parameter	An action that forces the PATROL Agent to run one or more parameters immediately, regardless of their polling cycle. Refreshing does not reset the polling cycle but gathers a new data point between polling cycles. Refresh parameter is also called “update” in PATROL 7.x.
reporting filter	The filter used by the PATROL Agent when transmitting events to consoles (event cache) from the event repository (located at the agent) for statistical reports.
response window	An input and output display for many KM menu commands that provides a customizable layout of the information (for example, the sort method for outputting system process IDs). <i>See also</i> system output window and task output window.
rule	An individual configuration item or variable.
ruleset	A collection of rules or configuration items.
run queue	<i>See</i> PATROL Agent run queue.
self-polling parameter	A standard parameter that starts a process that runs indefinitely. The started process periodically polls the resource that it is monitoring and emits a value that is captured by the PATROL Agent and published as the parameter value. Self-polling avoids the overhead of frequently starting external processes to collect a monitored value. A self-polling parameter differs from most other parameters that run scripts for a short time and then terminate until the next poll time.
session file	In PATROL 3.x and earlier, any of the files that are saved when changes are made and saved during the current PATROL Console session. A session file includes the session-1.km file, which contains changes to KMs loaded on your console, and the session-1.prefs file, which contains user preferences. Session files are replaced by management profiles in PATROL 7.x.

setup command	A command that is initiated by the PATROL Console and run by the PATROL Agent when the PATROL Console connects or reconnects to the agent. For example, a setup command can initialize an application log file to prepare it for monitoring. PATROL provides some setup commands for computer classes. Only a PATROL Console with developer functionality can add or change setup commands.
shortcut	An alias or copy of an object icon in the PATROL hierarchy.
simple discovery	A type of application discovery that uses simple pattern matching for identifying and monitoring files and processes.
SNMP	<i>See</i> Simple Network Management Protocol (SNMP).
Simple Network Management Protocol (SNMP)	A communications protocol that is supported by the PATROL Agent. SNMP allows network management systems to access PATROL Agents and allows PATROL Agents to monitor and manage SNMP devices.
SNMP trap	A condition which, when satisfied, results in an SNMP agent issuing a trap message to other SNMP agents and clients. Within the PATROL Agent, all events can be translated to SNMP traps and forwarded to SNMP managers.
snooze an alarm	To temporarily suspend an alarm so that a parameter does not exhibit an alarm state. During the user-set snooze period, the parameter continues to run commands and recovery actions, and the parameter icon appears to be in an OK state. <i>See also</i> deactivate a parameter and suspend a parameter.
Standard Event Catalog	A PATROL-provided collection of predefined event classes for all computer classes and application classes. To add, modify, or delete event classes and commands in the Standard Event Catalog, you must use a PATROL Console with developer functionality. <i>See also</i> event catalog and event class.

standard parameter	A type of parameter that collects and displays data and can also execute commands. A standard parameter is like a collector parameter and consumer parameter combined. <i>See also</i> collector parameter, consumer parameter, and parameter.
startup command	<i>See</i> setup command.
state	The condition of an object (computer instance, application instance, or parameter) monitored by PATROL. The most common states are OK, warning, and alarm. Object icons can show additional conditions. <i>See also</i> application state, computer state, parameter state, and state change action.
state Boolean	A parameter output style that represents the on or yes state of a monitored object as a check mark and the off or no state as the letter <i>x</i> . Parameters with this output style can have alerts (warning and alarm) and recovery actions. Numeric data output for the monitored object can be displayed as a graph. <i>See also</i> spotlight.
state change action	An action that is stored, maintained, and initiated by the PATROL Console when the console is notified by the PATROL Agent that a monitored object has changed state. The action, or command, executes on the computer on which the console is running, not the computer on which the agent is running.
static KM	A KM that is not loaded by the PATROL Agent before a PATROL Console with a loaded KM of the same name connects to the Agent. Once loaded by the agent, a static KM is never unloaded but continues to run as long as the agent runs, even if all PATROL Consoles with a registered interest disconnect from the PATROL Agent. If the PATROL Agent stops, static KMs will not be reloaded. <i>See also</i> disable an application, disable a KM and preloaded KM.

stoplight	A parameter output style that displays OK, warning, and alarm states as green, yellow, and red lights, respectively, on a traffic light. Parameters with this output style can have alerts (warning and alarm) and recovery actions. Numeric data output for the monitored object can be displayed as a graph. <i>See also</i> state Boolean.
suspend a parameter	To stop running a parameter for selected computers or application instances. Suspending a parameter stops parameter commands and recovery actions but does not delete the parameter icon from the application instance window and does not delete the parameter definition from the KM tree in PATROL Consoles for Microsoft Windows environments. A suspended parameter can be resumed at any time. You can suspend a parameter from its pop-up menu. <i>See also</i> deactivate a parameter and snooze an alarm.
system output window	A message window that displays the output of commands and tasks that the PATROL Console or the PATROL Agent execute on an instance. The window also displays error messages, commit status messages, and so forth. When the system output window contains unread messages in PATROL 3.x and earlier, the instance icon displays a yellow triangle for Windows; for Unix, it displays a blue screen with white text.
task	A command or group of commands that can execute on one object or several objects simultaneously. A task runs in the background and is not part of the PATROL Agent run queue; a task icon is displayed for each running task.
task output window	A window that contains command output generated by a task (for example, a KM menu command or a parameter warning or alarm). While executing, each task has its own icon, which usually appears in the PATROL interface or main window but may appear in an appropriate object window.
threshold	A point or points that define a range of values, outside of which a parameter is considered to be in a warning or alarm range.

unload a KM

To delete a KM from a PATROL Console session in order to stop monitoring the KM-defined objects on all computers. The KM files are not deleted from the directories on the PATROL Console or the PATROL Agent computers, and the PATROL Agent will continue to run the KM, collect parameter data, and run recovery actions until no connected console has the KM loaded. To prevent the PATROL Agent computer from collecting parameter data and running recovery actions for a KM, disable the KM. If a KM has been flagged as static, then it will not be unloaded. *See also* disable an application, disable a KM, preloaded KM, and static KM.

User Datagram Protocol (UDP)

In PATROL 3.x and earlier, a connectionless network protocol that allows the PATROL Console to connect to many agents simultaneously. TCP requires an open file for each connection, and the number of files that a process can have open is generally limited.

user preferences

The PATROL Console settings that designate the account that you want to use to connect to monitored host computers, prevent a console with developer functionality from downloading its version of a KM to a PATROL Agent upon connection, disable the commit process for a console with developer functionality, determine certain window and icon display characteristics, specify the event cache size, and indicate whether startup and shutdown commands are enabled. A PATROL Console with either developer or operator functionality can change user preferences.

version arbitration

In PATROL 3.x and earlier, the KM version comparison that PATROL makes when a PATROL Console connects to a PATROL Agent. By default, KM versions from PATROL Consoles with developer functionality are loaded rather than PATROL Agent KM versions, and PATROL Agent KM versions are loaded rather than KM versions from PATROL Consoles with operator functionality.

view filter

A filter that can be created in an event manager (PEM) and that screens events forwarded from PATROL Agents. Views can be created, stored, and reapplied to host computers.

warning	An indication that a parameter has returned a value that falls within the warning range. <i>See also</i> alarm.
wpconfig	<i>A feature of PATROL for Microsoft Windows only.</i> The graphical user interface utility for setting PATROL Agent configuration variables. The wpconfig utility can be accessed from a computer pop-up menu on a computer running a PATROL Agent or a computer running a PATROL Console with developer functionality. <i>See also</i> PATROL Agent configuration file and PATROL Agent configuration variable.
xpconfig	<i>A feature of PATROL for Unix only.</i> The graphical user interface utility for setting PATROL Agent configuration variables. You can access the xpconfig utility from an xterm session command line on a computer running a PATROL Agent or from a pop-up menu or an xterm session command line on a PATROL Console with developer functionality. <i>See also</i> PATROL Agent configuration file and PATROL Agent configuration variable.

Index

Symbols

__ANYINST__ 4-23

A

accessing

 KM application class menus 2-30
 menus 2-30

administrator account 2-4

Alert Actions 4-3

alert condition 4-11

Alert Messages 4-12

 Replacement Variables 4-12

Alert Settings 2-29, 4-3

alertLocalCommand 4-5

alertResend 4-5

alertResendOnInit 4-5

application classes 1-4

 AS_AVAILABILITY 1-4

 AS_EVENTSPRING 1-4

 hierarchy 1-5

 icons 1-4

architecture 1-2

arsCmdType 4-5

arsCommand 4-5

availability blackouts 4-20

B

blackout periods 1-2, 2-29

Blat 2-31

C

command-line interface A-1

command-line notification utilities

 Blat 2-31

 mailx 2-31

commands

 About 4-28, 4-30

 Acknowledge Open Events 4-29

 Add Target 4-18

 Admin 4-27

 Clear Alert Queues 4-27

 Alert Messages 4-12

 alert settings 4-3

 Alert Testing 4-29, 4-30

 Test ALARM 4-29, 4-30

 Test WARN 4-29, 4-30

 TEST_NOTIFY_EVENT 4-29, 4-30

 AS_AVAILABILITY application 4-30

 AS_EVENTSPRING application 4-28

 Availability 4-18

 Blackout Periods 4-16, 4-20

- Change Filter Type 4-24
 - Exclude 4-24
 - Include 4-24
- Checker Account 4-21
- Close Acknowledged Events 4-29
- COMPUTER menu 4-2
- Configuration DB 4-25
 - Delete Variables 4-25
 - Display Values 4-25
- Configure Notification Servers 4-8
- Custom Identifiers 4-17
- Edit Filter List 4-24
- Failover Settings 4-19
- Filtered Instance Report 4-25
- Identify Primary 4-20
- Instance Filtering 4-24
- Local Alert Settings 4-5
- Manage Events 4-28
- menu 4-1
- Notification Command 4-6
- Notification Server Settings 4-16
- Notification System 4-4
- Notification Targets 4-9
 - Custom Target 4-10
 - email 4-9
 - None 4-9
 - Pager Target 4-10
 - TT Target 4-11
- Overrides 4-17
- Parameter Settings 4-22, 4-25
- Ping Command 4-21
- Polltimes 4-23
- Recovery Action Command 4-6
- Recovery Action Command Type 4-7
- Recovery Action Output 4-27
- Refresh Parameters 4-29, 4-31
- Remote Alert Settings 4-8
- Remote Comm Settings 4-8
- Remote Target Setting 4-16
 - options 4-17
- Remove Primary 4-20
- Remove Targets 4-19
- Report Targets 4-21
- Reports 4-25, 4-29
- Send Reset On Init 4-7
- Status Flags 4-23
- Thresholds 4-22
- configuration
 - Add Target 4-18
 - alert actions 4-3
 - Alert Messages 4-12
 - Alert Resend 4-5
 - Blackout Periods 4-16, 4-21
 - Checker Account 4-22
 - Custom Identifiers 4-17
 - Custom Target 4-11
 - Edit Filter List 4-24
 - Email Target 4-10
 - identify notification servers 2-26
 - Identify Primary 4-20
 - information needed 2-26, 2-28
 - Notification Command 4-6
 - notification servers 2-36
 - Notification System 4-4
 - options 2-29
 - Overrides 4-18
 - Pager Target 4-10
 - Ping Command 4-21
 - Polltimes 4-23
 - Recovery Action Command 4-6
 - Recovery Action Command Type 4-7
 - remote agents 2-38
 - Remote Comm Settings 4-8
 - Remote Target Setting 4-17
 - Send Reset On Init 4-7
 - Status Flags 4-23
 - Thresholds 4-22
 - TT Target 4-11
 - Updated Flag 4-19
 - variables 3-1

configurationConfigure Notification Servers
4-8
conventions, document xvii

D

default PATROL account 2-4
dependencies 2-19, 2-20, 2-21, 2-22
dialog
 PARAMETER SETTINGS REPORT
 4-26
document conventions xvii

E

e-mail 1-2, 1-3, 3-4
event management rules 3-1
example
 e-mail rule 3-5
 PATROL objects 3-3
 rule inheritance 3-4

F

failover 1-4
file
 StdEvents.ctg 2-6

I

identify command-line notification utilities
2-31
installation procedures
 Custom 2-12
 migration 2-15
 planning 2-6
 Typical 2-9

instance name
 __ANYINST__ 4-23
interface
 command-line A-1

K

KM
 before you configure KM 2-29

L

license 2-3
loading KMs
 with the PATROL Central - Web Edition
 2-20
 with the PATROL Central -Windows
 Edition 2-19
 with the PATROL Console for Unix
 2-22
 with the PATROL Console for Windows
 2-21
loading Knowledge Modules 2-19
logon 2-4
logon account 2-4

M

mailx 2-31
menu commands 4-1
menus 2-30
messages
 rewording 1-2, 2-29

N

notification 1-2, 1-3

- server 1-4, 2-26
- targets 1-2
- utilities 2-31
- NOTIFY_EVENT A-1

O

- Object Name Filter 4-26
- online documentation xvi

P

- paging 1-2, 1-3
- Parameter Settings Report Formats 4-26
- parameters
 - AgentLoginDenied 5-2
 - AgentPingFailures 5-2
 - AvailabilityMonitorColl 5-2
 - default values 5-3
 - defaults
 - AgentLoginDenied 5-3
 - AgentPingFailures 5-3
 - AlertTest 5-3
 - AvailabilityMonitorColl 5-3
 - HostPingFailures 5-3
 - NotifyEvents 5-3
 - RefreshParamSettings 5-3
 - ResendAlertQueue 5-3
 - RetriggerEventQueue 5-3
 - SnmpPingFailures 5-3
 - HostPingFailures 5-2
 - list of 5-2
 - NotifyEvents 5-2
 - RefreshParamSettings 5-2
 - ResendAlertQueue 5-2
 - RetriggerEventQueue 5-2
 - set value B-4
 - SnmpPingFailures 5-2
 - summary 5-1

- parametersAlertTest 5-2
- PATROL account 2-4
- PATROL objects 3-1, 3-2
 - example 3-3
- Perl 2-32
- product
 - application classes 1-4
 - architecture 1-2
 - capabilities 1-1
 - components 1-1
 - features 1-2
- Property Filters 4-26

R

- recovery actions 1-2
- release notes xvi
- Reports 4-25, 4-29
 - all acknowledged notification events 4-29
 - all escalated notification events 4-29
 - all notification events 4-29
 - all Open notification events 4-29
 - Parameter Settings 4-25
- requirements
 - accounts 2-4
 - default PATROL account 2-4
 - license 2-3
 - logon account 2-4
 - PATROL security 2-4
 - system 2-3
- rewording of alert messages 2-29
- rules
 - Alert and Notification Settings B-2
 - alertLocalCommand B-4
 - alertResend B-4
 - alertResetOnInit B-5
 - alertSystem B-4
 - allowOverrides B-3
 - arsAction B-2

- arsCmdType B-2
- arsCommand B-2
- blackoutPeriod B-7
- customId B-3
- customTargetsLocal B-6
- e-mail 3-5
- emailTargetsLocal B-5
- event management 3-1
- inheritance 3-4
- loginDeniedIgnoredUsers B-3
- msgText B-6
- pagerTargetsLocal B-5
- reference B-1
- setParameterValue B-4
- spoolDirectory B-3
- ttTargetsLocal B-6
- useEnvOnlyForCmds B-3

S

- set parameter value B-4
- settings
 - active B-14
 - allowOperator B-15
 - appClassSettingsStatusFlag B-14
 - Application Class B-14
 - Availability Monitor B-10
 - Blackouts B-10
 - checkerAccount B-11
 - interval B-12
 - Menu Command Access B-15
 - Notification Server B-9
 - NOTIFICATION_SERVER B-8
 - nsRemoteTargetSetting B-9
 - Parameter B-12
 - paramSettingsStatusFlag B-12
 - pingCmd B-11
 - pingOkString B-11
 - Primary B-10, B-11
 - Remote Notification B-8

- RemoteAgentCommSettings B-8
- Targets B-10
- Updated B-11
- StdEvents.ctg file 2-6

T

- test notification 2-34
- THRESHOLDS B-13

U

- unloading KMs
 - with the PATROL Central - Web Edition 2-24
 - with the PATROL Central - Windows Edition 2-23
 - with the PATROL Console for Unix 2-25
 - with the PATROL Console for Windows 2-24
- Updated Flag 4-19

V

- variables
 - active B-14
 - allowOperator B-15
 - appClassSettingsStatusFlag B-14
 - Blackouts B-10
 - checkerAccount B-11
 - Interval B-12
 - NOTIFICATION_SERVER1 B-8
 - NOTIFICATION_SERVER2 B-8
 - nsRemoteTargetSetting B-9
 - paramSettingsStatusFlag B-12
 - pingCmd B-11
 - pingOkString B-11

Primary B-10, B-11
RemoteAgentCommSettings B-8
Targets B-10
Targets2 B-10
Updated B-11
verifying discovery 2-30

W

warning
 administrator account 2-4

STOP!

IMPORTANT INFORMATION - DO NOT INSTALL THIS PRODUCT UNLESS YOU HAVE READ ALL OF THE FOLLOWING MATERIAL

By clicking the YES or ACCEPT button below (when applicable), or by installing and using this Product or by having it installed and used on your behalf, You are taking affirmative action to signify that You are entering into a legal agreement and are agreeing to be bound by its terms, EVEN WITHOUT YOUR SIGNATURE. BMC is willing to license this Product to You ONLY if You are willing to accept all of these terms. CAREFULLY READ THIS AGREEMENT. If You DO NOT AGREE with its terms, DO NOT install or use this Product; press the NO or REJECT button below (when applicable) or promptly contact BMC or your BMC reseller and your money will be refunded if by such time You have already purchased a full-use License.

SOFTWARE LICENSE AGREEMENT FOR BMC PRODUCTS

SCOPE. This is a legally binding Software License Agreement (“**License**”) between You (either an individual or an entity) and BMC pertaining to the original computer files (including all computer programs and data stored in such files) contained in the enclosed Media (as defined below) or made accessible to You for electronic delivery, if as a prerequisite to such accessibility You are required to indicate your acceptance of the terms of this License, and all whole or partial copies thereof, including modified copies and portions merged into other programs (collectively, the “**Software**”). “**Documentation**” means the related hard-copy or electronically reproducible technical documents furnished in association with the Software, “**Media**” means the original BMC-supplied physical materials (if any) containing the Software and/or Documentation, “**Product**” means collectively the Media, Software, and Documentation, and all Product updates subsequently provided to You, and “**You**” means the owner or lessee of the hardware on which the Software is installed and/or used. “**BMC**” means BMC Software Distribution, Inc. unless You are located in one of the following regions, in which case “**BMC**” refers to the following indicated BMC Software, Inc. subsidiary: (i) Europe, Middle East or Africa --BMC Software Distribution, B.V., (ii) Asia/Pacific -- BMC Software Asia Pacific Pte Ltd., (iii) Brazil -- BMC Software do Brazil, or (iv) Japan -- BMC Software K.K. **If You enter into a separate, written software license agreement signed by both You and BMC or your authorized BMC reseller granting to you the rights to install and use this Product, then the terms of that separate, signed agreement will apply and this License is void.**

FULL-USE LICENSE. Subject to these terms and payment of the applicable license fees, BMC grants You this non-exclusive License to install and use one copy of the Software for your internal use on the number(s) and type(s) of servers or workstations for which You have paid or agreed to pay to BMC or your BMC reseller the appropriate license fee. If your license fee entitles You only to a License having a limited term, then the duration of this License is limited to that term; otherwise this License is perpetual, subject to the termination provisions below.

TRIAL LICENSE. If You have not paid or agreed to pay to BMC or your BMC Reseller the appropriate license fees for a full use license, then, **NOTWITHSTANDING ANYTHING TO THE CONTRARY CONTAINED IN THIS LICENSE:** (i) this License consists of a non-exclusive evaluation license (“Trial License”) to use the Product for a limited time (“Trial Period”) only for evaluation; (ii) during the Trial Period, You may not use the Software for development, commercial, production, database management or other purposes than those expressly permitted in clause (i) immediately above; and (iii) your use of the Product is on an **AS IS** basis, and **BMC, ITS RESELLERS AND LICENSORS GRANT NO WARRANTIES OR CONDITIONS (INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE) TO YOU AND ACCEPT NO LIABILITY WHATSOEVER RESULTING FROM THE USE OF THIS PRODUCT UNDER THIS TRIAL LICENSE.** If You use this Product for other than evaluation purposes or wish to continue using it after the Trial Period, you must purchase a full-use license. When the Trial Period ends, your right to use this Product automatically expires, though in certain cases You may be able to extend the term of the Trial Period by request. Contact BMC or your BMC reseller for details.

TERM AND TERMINATION. This License takes effect on the first to occur of the date of shipment or accessibility to You for electronic delivery, as applicable (the “**Product Effective Date**”). You may terminate this License at any time for any reason by written notice to BMC or your BMC reseller. This License and your right to use the Product will terminate automatically with or without notice by BMC if You fail to comply with any material term of this License. Upon termination, You must erase or destroy all components of the Product including all copies of the Software, and stop using or accessing the Software. Provisions concerning Title and Copyright, Restrictions (or Restricted Rights, if You are a U.S. Government entity) or limiting BMC’s liability or responsibility shall survive any such termination.

TITLE AND COPYRIGHT; RESTRICTIONS. All title and copyrights in and to the Product, including but not limited to all modifications thereto, are owned by BMC and/or its affiliates and licensors, and are protected by both United States copyright law and applicable international copyright treaties. You will not claim or assert title to or ownership of the Product. To the extent expressly permitted by applicable law or treaty notwithstanding this limitation, You may copy the Software only for backup or archival purposes, or as an essential step in utilizing the Software, but for no other purpose. You will not remove or alter any copyright or proprietary notice from

copies of the Product. You acknowledge that the Product contains valuable trade secrets of BMC and/or its affiliates and licensors. Except in accordance with the terms of this License, You agree (a) not to decompile, disassemble, reverse engineer or otherwise attempt to derive the Software's source code from object code except to the extent expressly permitted by applicable law or treaty despite this limitation; (b) not to sell, rent, lease, license, sublicense, display, modify, time share, outsource or otherwise transfer the Product to, or permit the use of this Product by, any third party; and (c) to use reasonable care and protection to prevent the unauthorized use, copying, publication or dissemination of the Product and BMC confidential information learned from your use of the Product. **You will not export or re-export any Product without both the written consent of BMC and the appropriate U.S. and/ or foreign government license(s) or license exception(s).** Any programs, utilities, modules or other software or documentation created, developed, modified or enhanced by or for You using this Product shall likewise be subject to these restrictions. BMC has the right to obtain injunctive relief against any actual or threatened violation of these restrictions, in addition to any other available remedies. Additional restrictions may apply to certain files, programs or data supplied by third parties and embedded in the Product; consult the Product installation instructions or Release Notes for details.

LIMITED WARRANTY AND CONDITION. If You have purchased a Full-Use License, BMC warrants that (i) the Media will be, under normal use, free from physical defects, and (ii) for a period of ninety (90) days from the Product Effective Date, the Product will perform in substantial accordance with the operating specifications contained in the Documentation that is most current at the Product Effective Date. BMC's entire liability and your exclusive remedy under this provision will be for BMC to use reasonable best efforts to remedy defects covered by this warranty and condition within a reasonable period of time or, at BMC's option, either to replace the defective Product or to refund the amount paid by You to license the use of the Product. BMC and its suppliers do not warrant that the Product will satisfy your requirements, that the operation of the Product will be uninterrupted or error free, or that all software defects can be corrected. This warranty and condition shall not apply if: (i) the Product is not used in accordance with BMC's instructions, (ii) a Product defect has been caused by any of your or a third party's malfunctioning equipment, (iii) any other cause within your control causes the Product to malfunction, or (iv) You have made modifications to the Product not expressly authorized in writing by BMC. No employee, agent or representative of BMC has authority to bind BMC to any oral representations, warranties or conditions concerning the Product. **THIS WARRANTY AND CONDITION IS IN LIEU OF ALL OTHER WARRANTIES AND CONDITIONS. THERE ARE NO OTHER EXPRESS OR IMPLIED WARRANTIES OR CONDITIONS, INCLUDING THOSE OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, REGARDING THIS LICENSE OR ANY PRODUCT LICENSED HEREUNDER. THIS PARAGRAPH SHALL NOT APPLY TO A TRIAL LICENSE.** Additional support and maintenance may be available for an additional charge; contact BMC or your BMC reseller for details.

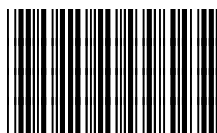
LIMITATION OF LIABILITY. Except as stated in the next succeeding paragraph, BMC's and your BMC reseller's total liability for all damages in connection with this License is limited to the price paid for the License. **IN NO EVENT SHALL BMC BE LIABLE FOR ANY CONSEQUENTIAL, SPECIAL, INCIDENTAL, PUNITIVE OR INDIRECT DAMAGES OF ANY KIND ARISING OUT OF THE USE OF THIS PRODUCT (SUCH AS LOSS OF PROFITS, GOODWILL, BUSINESS, DATA OR COMPUTER TIME, OR THE COSTS OF RECREATING LOST DATA), EVEN IF BMC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.** Some jurisdictions do not permit the limitation of consequential damages so the above limitation may not apply.

INDEMNIFICATION FOR INFRINGEMENT. BMC will defend or settle, at its own expense, any claim against You by a third party asserting that your use of the Product within the scope of this License violates such third party's patent, copyright, trademark, trade secret or other proprietary rights, and will indemnify You against any damages finally awarded against You arising out of such claim. However, You must promptly notify BMC in writing after first receiving notice of any such claim, and BMC will have sole control of the defense of any action and all negotiations for its settlement or compromise, with your reasonable assistance. BMC will not be liable for any costs or expenditures incurred by You without BMC's prior written consent. If an order is obtained against your use of the Product by reason of any claimed infringement, or if in BMC's opinion the Product is likely to become the subject of such a claim, BMC will at its option and expense either (i) procure for You the right to continue using the product, or (ii) modify or replace the Product with a compatible, functionally equivalent, non-infringing Product, or (iii) if neither (i) nor (ii) is practicable, issue to You a pro-rata refund of your paid license fee(s) proportionate to the number of months remaining in the 36 month period following the Product Effective Date. This paragraph sets forth your only remedies and the total liability to You of BMC, its resellers and licensors arising out of such claims.

GENERAL. This License is the entire understanding between You and BMC concerning this License and may be modified only in a mutually signed writing between You and BMC. If any part of it is invalid or unenforceable, that part will be construed, limited, modified, or severed so as to eliminate its invalidity or unenforceability. This License will be governed by and interpreted under the laws of the jurisdiction named below, without regard to conflicts of law principles, depending on which BMC Software, Inc. subsidiary is the party to this License: (i) BMC Software Distribution, Inc. - the State of Texas, U.S.A., (ii) BMC Software Distribution, B.V. - The Netherlands, (iii) BMC Software Asia Pacific Pte Ltd. -- Singapore (iv) BMC Software do Brazil -- Brazil, or (v) BMC Software K.K. -- Japan. Any person who accepts or signs changes to the terms of this License promises that they have read and understood these terms, that they have the authority to accept on your behalf and legally obligate You to this License. Under local law and treaties, the restrictions and limitations of this License may not apply to You; You may have other rights and remedies, and be subject to other restrictions and limitations.

U.S. GOVERNMENT RESTRICTED RIGHTS. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b) or DFARS 227.7202, as amended from time to time. Contractor/Manufacturer is BMC Software, Inc., 2101 CityWest Blvd., Houston, TX 77042-2827, USA. Any contract notices should be sent to this address.

Notes



23384